Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning
KR meets Machine Learning and Explanation

# Precise and Efficient Model-Agnostic Explanations

**Jairo A. Lefebre-Lobaina**[1] , **Maria Vanina Martinez**[1] , **Joao Marques-Silva**[2]
[1]Artificial Intelligence Research Institute (IIIA), CSIC, Spain
[2]ICREA & Univ. Lleida, Spain
{jairo.lefebre, vmartinez}@iiia.csic.es, jpms@icrea.cat

**Abstract**

Logic-based eXplainable Artificial Intelligence (XAI) represents a rigorous alternative to non-symbolic XAI. However, one critical limitation of logic-based explanations is the complexity of reasoning about machine learning (ML) models. Sample-based explanations represent a rigorous, model-agnostic, but also scalable alternative to model-based explanations. Whereas finding one sample-based explanation can be done in polynomial time, the computation of a smallest explanation is computationally hard. This paper develops CovAXp, a novel heuristic method for the computation of small sample-based explanations. The proposal employs a feature-coverage heuristic within a pruned depth-first search that prioritizes features maximizing rule coverage within the sample space. Experimental evaluation shows that CovAXp achieves near-minimal explanation cardinality (mean length 2.48 versus the optimal 2.36), sub-second execution times, and perfect true negative rate, while offering a tunable trade-off between coverage and rule compactness.

## 1 Introduction

Explainable Artificial Intelligence (XAI) aims to provide human-interpretable justifications for machine learning (ML) predictions – a critical requirement for trustworthy AI amid increasing regulatory scrutiny (Chamola et al. 2023). While model-agnostic methods are widely adopted for their versatility in treating ML models as black boxes, they lack formal correctness guarantees and often yield explanations inconsistent with both the target model and training data (Marques-Silva and Ignatiev 2022).

These limitations have motivated interest in logic-based XAI, which offers rigorous, model-driven explanations through logical encodings and automated reasoning (Ignatiev 2020). However, logic-based approaches face significant challenges: (1) computational scalability for complex models, (2) unwieldy explanation sizes, and (3) dependency on complete logical encodings of ML models. The latter requirement proves particularly problematic for some models (e.g., kernel-based SVMs) or those containing proprietary/protected data where full disclosure is infeasible (Darwiche 2023).

Model-agnostic explanation methods can be categorized into two distinct paradigms: *feature importance attribution* methods like LIME (Ribeiro, Singh, and Guestrin 2016) and SHAP (Lundberg and Lee 2017), which assign significance scores to input features to rationalize predictions; and *feature selection* methods like Anchors (Ribeiro, Singh, and Guestrin 2018) , GreedyCF (Geng, Schleich, and Suciu 2022), and OptConsistentRule (Rudin and Shaposhnik 2023), which identify a subset-minimal feature set that determine predictions through human-interpretable rules.

To overcome the need for model internals, recent work on sample-based formal explanations (sbFXAI) (Cooper and Amgoud 2023; Amgoud, Cooper, and Debbaoui 2024; Marques-Silva, Lefebre-Lobaina, and Martinez 2025) presents a model agnostic feature selection approach with the same guarantees of logic-based XAI with respect to a sampling made to the model. In (Marques-Silva, Lefebre-Lobaina, and Martinez 2025) a polynomial-time computation of rigorous explanations from sampled data (we will call it FastAXp) was established, and in (Rudin and Shaposhnik 2023) it was proposed a way to compute rigorous explanations of minimal cardinality. While these approaches address the scalability limitations of pure logic-based approaches, among the different feature selection algorithms significant challenges persist in balancing three competing objectives: (1) minimal explanation cardinality to improve interpretability, (2) computational efficiency, and (3) consistency guarantees.

In classification tasks, particularly those with high-stakes negative outcomes (e.g., medical diagnostics or security systems), the True Negative Rate (TNR), often called specificity, becomes critical. TNR quantifies a model's reliability in safeguarding against false positives – minimizing erroneous identifications where 'absence' carries significant consequences (Mohammed and Wagner 2014; Han, Pei, and Tong 2022). Existing solutions have some undesirable trade-offs: Anchors achieves polynomial runtime but sacrifices TNR and minimality; GreedyCF ensures TNR but suffers from redundancy and the running time may be a problem for complex datasets; OptConsistentRule has TNR guarantees and ensures minimality but tackles an NP-hard problem; FastAXp is a polynomial time approach with TNR guarantees but the rule length may still be a problem.

This paper introduces *CovAXp*, a novel sample-based explanation algorithm that bridges the rigor of logic-based XAI with the practicality of model-agnostic approaches. Unlike existing methods, CovAXp employs a heuristic

search process that prioritizes features, maximizing rule coverage within the sample space.

This strategic design achieves five key advantages that balance efficiency, formal guarantees, and interpretability. First, our proposal computes the explanations substantially faster than searching for cardinality-minimal explanations. Second, it generates more compact explanations than those produced by comparable approaches, increasing the "perceived interpretability" of generated rules (Lakkaraju, Bach, and Leskovec 2016). Third, the method offers a formal guarantee that the generated explanations are consistent with the provided sample, ensuring soundness in practical applications. Fourth, it allows us to parameterize the coverage to increase perceived interpretability by allowing a tradeoff between the amount of false positive instances to achieve shorter and more comprehensible rules, making the resulting explanations easier to understand and apply in real-world decision-making contexts. Lastly, experimental evaluation on real-world datasets demonstrates CovAXp's significant advantages over state-of-the-art baselines in both computational efficiency and explanation quality.

The rest of this work is structured as follows: The 'Basic Concepts and Related Work' section formalizes classification problems and sample-based explanations, establishes the hitting set duality for sample-based explanations, and critically analyzes limitations of existing approaches through concrete examples. Next, 'Coverage Driven Explanations' introduces our algorithm CovAXp, detailing its search process, formal guarantees, and true negative rate-length tradeoff mechanism. The 'Experimental Evaluation' section validates CovAXp's performance across 35 real-world datasets, demonstrating order-of-magnitude speedups versus baselines while maintaining near-minimal explanation length. Finally, the Conclusion' synthesizes key findings and outlines future work.

## 2 Basic Concepts and Related Work

This section establishes the formal foundation and practical context of our work. We begin by formalizing classification problems and sample-based explanations, including the critical duality relationship between abductive and contrastive explanations. We then analyze state-of-the-art explanation methods—Anchors, GreedyCF, OptConsistentRule and FastAXp—through both theoretical and empirical lenses. Using a running example (Table 2), we show how existing approaches either lose rigor, minimality, or efficiency. These limitations motivate our proposed algorithm, CovAXp, which addresses these tradeoffs.

**Classification Problem and Explanation Quality Evaluation.** A classification problem is defined by a feature set $\mathcal{F} = \{1, \ldots, n\}$ with domains $\mathbf{D}_i$, forming a feature space $\mathbf{F} = \prod_{i=1}^{n} \mathbf{D}_i$. A classifier $\mathcal{M} = (\mathcal{F}, \mathbf{F}, \kappa, \mathcal{K})$ implements a non-constant function $\kappa : \mathbf{F} \to \mathcal{K}$ mapping to classes $\mathcal{K} = \{c_1, \ldots, c_K\}$. For logic-based XAI, we explain predictions for instances $(\mathbf{v}, c)$ where $\mathbf{v} \in \mathbf{F}$ and $c = \kappa(\mathbf{v})$.

The quality assessment of such classifiers relies fundamentally on confusion matrix analysis, which categorizes model predictions into four critical outcomes (Fawcett 2006): true positives (TP) when correctly predicts the expected class, false positives (FP) when incorrectly predicts the expeted class when the actual class is another, false negatives (FN) when incorrectly predicts other class class when the actual class is the expected, and true negatives (TN) when correctly predicts different class from the expected.

| Measure | Name | Definition |
|---|---|---|
| Accuracy | acc | $\frac{TP+TN}{TP+FP+TN+FN}$ |
| Precision | prec | $\frac{TP}{TP+FP}$ |
| True positive rate | TPR | $\frac{TP}{TP+FN}$ |
| True negative rate | TNR | $\frac{TN}{TN+FP}$ |

Table 1: Selection of standard measures of performance of a classifier.

From these core components, several essential metrics emerge to evaluate the predictions of a classifier. Table 1 presents the measures used in this work to evaluate the quality of an explanation rule evaluation for a classification black box model.

On the other hand, metrics like fidelity, comprehensibility, and stability are commonly used criteria for assessing the quality of explanation methods (Alangari et al. 2023). Each captures a distinct property that affects how well an explanation aligns with the underlying model and how useful it is for human reasoning. In this work the following quantitative evaluations metrics are used.

- **Fidelity**: Measures how accurately an explanation reflects the true decision-making behavior of the target model. Fidelity is often quantified through accuracy agreement metrics, evaluating the match between explanation and explained model (Bastani, Kim, and Bastani 2017).
- **Comprehensibility**: Evaluates how easily a human can interpret and reason with an explanation and is also related with brevity, since more concise explanations reduce cognitive load. In the context of rule-based explanations, it is measured by the number of rules, number of conditions, consistency in the rule set and the redundancy degree (Huysmans et al. 2011).
- **Stability**: Measures the robustness of an explanation to small perturbations in the input (Karimi 2025).

**Sample-Based Explanations.** Sample-based explanations are similar to logic-based explanations (Cooper and Amgoud 2023; Marques-Silva, Lefebre-Lobaina, and Martinez 2025), with the difference that the explanation rigor is restricted to some sample $\mathbf{S} \subseteq \mathbf{F}$. Given a sample $\mathbf{S}$, a *sample-based weak abductive explanation* (*sbWAXp*) is a feature subset $\mathcal{X} \subseteq \mathcal{F}$ satisfying:

$$\forall(\mathbf{x} \in \mathbf{S}).\ (\wedge_{i \in \mathcal{X}}(x_i = v_i)) \rightarrow (\kappa(\mathbf{x}) = \kappa(\mathbf{v})) \quad (1)$$

A subset-minimal *sbWAXp* is not considered *weak* since it is irreducible, hence named *sample-based AXp* (*sbAXp*).

When $\mathbf{S} = \mathbf{F}$, *sbAXps* correspond to exact *AXps*. Analogously, a *sample-based weak contrastive explanation* (*sbWCXp*) is a subset $\mathcal{Y} \subseteq \mathcal{F}$ satisfying:

$$\exists(\mathbf{x} \in \mathbf{S}).\ \big(\wedge_{i \in \mathcal{F} \setminus \mathcal{Y}}(x_i = v_i)\big) \wedge (\kappa(\mathbf{x}) \neq \kappa(\mathbf{v})) \qquad (2)$$

with subset-minimal *sbWCXps* termed *sample-based CXps* (*sbCXps*). In the rest of this work, subset-minimal explanations are also referred to as *irreducible* explanations.

To compute all *sbWCXps*, we define a per-feature similarity function $sf_i : \mathbf{D}_i \times \mathbf{D}_i \rightarrow \{0, 1\}$ for each feature $i \in \{1, \ldots, n\}$, where $sf_i(a, b) = 1$ indicates value similarity and 0 indicates dissimilarity. It is important to highlight that $sf_i$ does not assume that $\mathbf{D}_i$ belongs to any space or distribution. A very commonly used function for discrete $\mathbf{D}_i$ is strict equality, for continuous (potentially infinite) $\mathbf{D}_i$ a discretization process can be carried out and compare discrete partitions or use a function like:

$$sf_i(a, b) = \begin{cases} 1 & \text{if } |a - b| < \alpha \\ 0 & \text{otherwise} \end{cases}$$

Being $\alpha$ a minimum acceptable difference to consider the values as similar. For any pair of instances from different classes, a *sbWCXp* is generated by computing the negation of $sf_i$ for each feature, i.e. $1 - sf_i$.

The sample-based formal explanability framework is closely related to Logic Analysis of Data (Crama, Hammer, and Ibaraki 1988) and Testor Theory (Lazo-Cortes, Ruiz-Shulcloper, and Alba-Cabrera 2001). A *sbAXp* corresponds to an *Irreducible (or Typical) Testor* from Testor Theory. The Testor Theory is an active research line that has been applied to different domains (Alvarado-Moreira and Ibarra-Fiallo 2024; Pons-Porrata, Gil-García, and Berlanga-Llavori 2007). It has explored diverse similarity functions and is not restricted to strict equality (Lazo-Cortes, Ruiz-Shulcloper, and Alba-Cabrera 2001), this provides a foundation for extending sample-based formal explanations to richer feature spaces.

**Hitting Set Duality and Explanations.** Several relevant problems can be translated to a Minimal Hitting Set (MHS) problem. The MHS duality principle provides a theoretical foundation for computing abductive explanations, where *AXps* correspond to MHSs of all *CXps* and vice versa (Ignatiev et al. 2020). The search of all MHSs is an NP-hard problem, while the matrix representation of *sbWCXps* remains polynomial-sized relative to the sample, finding minimal-size hitting sets is NP-hard, and existing approaches have worst-case exponential time (Gainer-Dewar and Vera-Licona 2017). As a result, one recurrent goal in the field is to devise heuristic methods that provide in practice (and in polynomial-time) good quality solutions to the minimum-size hitting set problem. It is also clear that changing $\mathbf{F}$ to $\mathbf{S}$ does not affect MHS duality, *sbAXps* correspond to MHSs of all *sbCXps* (Geng, Schleich, and Suciu 2022) and vice versa.

**Proposition 1** (MHS Duality in explanations (Marques-Silva, Lefebre-Lobaina, and Martinez 2025))**.** *Each sbAXp is a minimal hitting set of all sbCXps, and each sbCXp is a minimal hitting set of all sbAXps.*

| row # | A | B | C | D | E | $\kappa(\cdot)$ |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 1 | 1 | 1 | 0 |
| 3 | 1 | 1 | 0 | 1 | 1 | 0 |
| 4 | 0 | 1 | 1 | 1 | 0 | 0 |
| 5 | 1 | 0 | 1 | 0 | 1 | 0 |
| 6 | 1 | 0 | 1 | 0 | 0 | 0 |
| 7 | 1 | 0 | 1 | 1 | 1 | 1 |
| 8 | 1 | 1 | 0 | 0 | 1 | 1 |
| 9 | 1 | 1 | 1 | 1 | 1 | 1 |

Table 2: Sampling dataset $\mathbf{S}$, our target is $(\mathbf{v}_9, c_9)$.

**Related Work.** Given an instance, i.e. a pair $(\mathbf{v}_j, c_j)$, where $\mathbf{v}_j \in \mathbf{F}$, and $c_j$ is the model's prediction for input $\mathbf{v}_j$, explanation by feature selection often consists on finding a rule $R$ of the form IF *cond* THEN *predict* $c_j$. In general, the interpretation of the rule is as follows: given some input to a machine learning (ML) model, the prediction is $c_j$ as long as the condition *cond* holds for any possible input. Several approaches have been proposed over the years for learning rule sets (or lists) from datasets in different contexts (Zaki 2000; Wang, He, and Han 2003; Yin and Han 2003; Yang, Yang, and Cohen 2017; Yang et al. 2021; Yang, Yang, and Sun 2024). In recent years, several approaches have been developed and used some of these techniques to explain the behavior of a ML model. Due to the impracticality of exhaustively querying the feature space of the problem ($\mathbf{F}$), these methods typically sample the ML model and explain it based on the sampling. We use the sampling ($\mathbf{S}$) shown in Table 2 throughout this analysis.

Table 2 shows a sample $\mathbf{S}$ from a classifier function $\kappa$. We use this model and dataset to demonstrate the behavior of sample-based explanation algorithms, with $(\mathbf{v}_9, c_9) = ((1, 1, 1, 1, 1), 1)$ as the target instance to explain. In this context a rule $R$ is Maximally Consistent, if it is able to exclude all instances that do not belong to the target class (TNR=1).

Anchors (Ribeiro, Singh, and Guestrin 2018) generates explanations through feature selection by sampling the model using a distribution derived from training data. Given an instance-prediction pair $(\mathbf{v}, c)$, it produces an *anchor*: a high-precision rule of the form IF IF *condition* THEN predict $c$.

By design, Anchors employs custom precision and coverage metrics that differ from standard rule-evaluation measures (Fürnkranz, Gamberger, and Lavrač 2012). This distinction arises because Anchors dynamically generates samples during the explanation process, whereas traditional metrics for classifiers (and consequently for rule quality) assume a fixed evaluation dataset. For consistent evaluation, in this work we will use the conventional definitions of accuracy and true negative rate (TNR).

Anchors dynamically generates its own samples during the explanation process that are different from $\mathbf{S}$. The sampling generated is associated to a *perturbation distribution*. However, given that the algorithm only pay attention to instances with the same prediction of the explained instance

| row # | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 1 |
| 2 | 1 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 1 | 0 | 0 |
| 4 | 1 | 0 | 0 | 0 | 1 |
| 5 | 0 | 1 | 0 | 1 | 0 |
| 6 | 0 | 1 | 0 | 1 | 1 |

Table 3: $sbWCXp$ matrix for instance $(\mathbf{v}_9, c_9)$

cannot provide warranties of excluding instances that do not belong to the class. Just using the sampling provided in Table 2 applying Anchors to $(\mathbf{v}_9, c_9)$ will result in rules like $(A = 1) \rightarrow 1$ or $(E = 1) \rightarrow 1$, that are able to cover all instances with the same category and at the same time fail to exclude several instances that belong to the other class. Even if we train a Random Forest Classifier with that sampling and use Anchors to explain the same instance with a sampling size of 100 yields:

$$R_{\text{Anchor}} = (A = 1 \land B = 1 \land E = 1) \rightarrow 1$$

Evaluating this rule over the provided sampling in Table 2 it reveals imperfect exclusion of true negative instances. For example, the rule fails to exclude the instance $\mathbf{v}_3 = (1, 1, 0, 1, 1)$, also predicted by the model $c_3 = 0$.

Alternative approaches (mentioned below) use contrastive explanations from the sample ($sbWCXps$) to guide rule construction. This strategy allows to ensure that the generated rule $R$ excludes all instances in $\mathbf{S}$ not belonging to the same class of the explanation target. This strategy has been used in earlier work and it allows to build maximally consistent rules ($\text{TNR}(R) = 1.0$). As was mentioned before, to build the set of $sbWCXp$ is required to define a Boolean output comparison function per feature $sf_i$ mapping if two values in $\mathbf{D}_i$ are different (represented with value 1) or similar (represented with value 0), Table 3 shows the $sbWCXp$ matrix for the instance $(\mathbf{v}_9, c_9)$ when using strict equality as $sf_i$ for all features.

OptConsistentRule (from now on we will call it MinAXp) (Rudin and Shaposhnik 2023) also finds rules with maximal consistency using a sampling. It also provides guarantees for finding optimal (minimal length and maximally consistent) rules. However, the inherent computational complexity of this problem hinders the applicability to high-dimensional or large-scale data. For $(\mathbf{v}_9, c_9)$ it yields:

$$R_{\text{MinAXp}} = (A = 1 \land C = 1 \land D = 1) \rightarrow 1$$

To measure rule quality, we use the notion of *Global Consistency*: for every instance in $\mathbf{S}$ that satisfies the explanation rule, the prediction made by the model will be the same. Also, the authors coded the problem of finding minimal length and maximally consistent rules as a Mixed Integer Linear Programming (MILP) problem using Gurobi[1] as the backend. We opted for an open-source solution, and so we encoded the problem as a Boolean Satisfiability Problem (SAT) and used a SAT-solver as oracle without restricting the execution time budget (Marques-Silva, Lefebre-Lobaina, and Martinez 2025).

GreedyCF (Geng, Schleich, and Suciu 2022) uses a greedy approach based on hitting set duality. It incrementally constructs rule $R$ by selecting max number of $|k|_{\max}$ uncovered rows from the $sbWCXp$ matrix. If such rows do not exist, $R$ is maximally consistent. Otherwise, for the submatrix formed by selecting up to $k$ rows and excluding the components already in $R$, the algorithm uses the hitting set duality property to search for all rules (*sbAXps*). Next, the candidates are then sorted by length. The algorithm selects a shortest length rule to combine with $R$.

Consider GreedyCF applied to $(\mathbf{v}_9, c_9)$ using the $sbWCXps$ from Table 3, having the feature $B$ in the initial rule ($R_0$) and $|k|_{\max} = 2$ yields:

$$R_{\text{GreedyCF}} = (B = 1 \land A = 1 \land D = 1 \land C = 1) \rightarrow 1$$

GreedyCF tries to maximize the *Global Consistency* during the search process, but it cannot provide warranties of finding irreducible abductive explanations ($sbAXp$) returning a *sbWAXp* in some cases. Note that the resulting rule $R_{\text{GreedyCF}}$ is not a $sbAXp$: removing $B = 1$ preserves consistency ($A = 1 \land D = 1 \land C = 1$ suffices). Increasing the value of $|k|_{\max}$ is a possible solution, but the number of all minimal hitting sets is worst-case exponential on the number of features. Moreover, the enumeration of minimal hitting sets can be achieved with quasi-polynomial time delay, although in practice other solutions are preferred (Fredman and Khachiyan 1996).

FastAXp (Marques-Silva, Lefebre-Lobaina, and Martinez 2025) provides an irreducible rule efficiently, but without cardinality-minimal guarantees. For our example, it yields:

$$R_{\text{FastAXp}} = (A = 1 \land B = 1 \land C = 1 \land E = 1) \rightarrow 1$$

The next section introduces CovAXp, our algorithm for finding maximally consistent rules. Like GreedyCF, it builds rules incrementally, but prioritizes features that maximize the amount of rows hit by the explanation. Crucially, in contrast with GreedyCF, CovAXp produces irreducible ($sbAXp$) explanations and the search strategy allows building rules of reduced length.

## 3 Coverage Driven Explanations

This section presents the CovAXp algorithm for computing compact explanations while maintaining the maximal TNR guarantees of GreedyCF, MinAXp and FastAXp approaches. As previously established, the $sbWCXp$ matrix is constructed using a similarity function *sf* that compares the target instance against instances of different classes, encoding feature differences as 1s and similarities as 0s. Complementarily, we define a sample-based similarity matrix ($sbSim$) by comparing the target instance against same-class instances using $sf_i$. Table 4 shows the $sbSim$ matrix for the target instance $((1, 1, 1, 1, 1), 1)$.

The algorithm uses a *feature coverage* metric to quantify how comprehensively a feature $i$ explains the behavior of the model in $\mathbf{S}$. It measures $i$'s *explanatory footprint* by counting its appearances in $sbWCXp \cup sbSim$. Higher coverage indicates $i$ consistently influences predictions across

[1] https://www.gurobi.com/

| row # | A | B | C | D | E |
|---|---|---|---|---|---|
| 7 | 1 | 0 | 1 | 1 | 1 |
| 8 | 1 | 1 | 0 | 0 | 1 |
| 9 | 1 | 1 | 1 | 1 | 1 |

Table 4: $sbSim$ matrix

diverse cases. This strategy is a way to include the precision provided from contrastive instances and the increase of accuracy provided by the instances that belongs to the same class of the explained instance.

**Definition 2** (Feature Coverage)**.** Given a contrastive explanations matrix $sbWCXp$ and a similarity matrix $sbSim$, the *feature coverage* metric quantifies the coverage of feature $i$ in $\mathbf{S}$, based on the weighted sum of the 1 values of $i$ in $sbWCXp$ and $sbSim$.

The *feature coverage* $fc(i)$ for a feature $i$ is defined as:

$$fc(i) = \sum_{k=1}^{m} sbWCXp[k,i] + \frac{1}{m'} \sum_{k=1}^{m'} sbSim[k,i]$$

where $m = |sbWCXp|$, $m' = |sbSim|$ denote row counts, and $sbWCXp[k,i]$, $sbSim[k,i]$ represent feature $i$'s value in the $k$-th row of each matrix.

The feature coverage metric $fc(i)$ serves as a prioritization heuristic in CovAXp, guiding the selection of candidate features during rule construction. Note that other functions may be defined by changing the priority used to select features. Although this heuristic may influence the length of generated rules, it does not affect the algorithm's formal guarantees of maximal consistency over $\mathbf{S}$. The search is carried out traversing the $sbWCXp$ matrix and stops when the number of *false negative* instances is reduced to a desired portion (ideally 0). The role of the proposed heuristic function is to reduce the amount *false positive* in the same search process.

**Coverage Explanation Function.** Let $wxp$ be a matrix of sample-based weak contrastive explanations, $fcw$ a feature-weight function, $tnr \in [0,1]$ a desired true-negative rate, and $ml \in \mathbb{N}$ a maximum rule length. Let $ud \subseteq \mathcal{F}$ denote the current unitary description and $rf \subseteq \mathcal{R}$ the current set of uncovered rows. We define the iterative search procedure CovAXp in Algorithm 1.

CovAXp is inspired on the *RegularSerch* algorithm for testor computation (Lefebre-Lobaina and Shulcloper 2023). CovAXp use the same strategy and core properties, but it is focused in the construction a single $sbAXp$. It uses a more precise validation strategy for pruning subspaces and adds a greedy sorting of the feature sub-spaces, evaluating first features that are more likely to contribute to the result. The algorithm maintains two main structures across the iterations: (1) a *unitary description* $ud$, mapping each selected feature to the rows where it is the sole covering feature (ensuring irreducibility); and (2) a *row filter* $rf \subseteq \mathcal{R}$, tracking uncovered rows in $sbWCXp$, where $\mathcal{R}$ denotes the set of all row indices. Feature selection is guided by a coverage-based ordering generated by FilterSort$(wxp, fcw, rf)$, which returns the features not yet in the current explanation, sorted descending by their coverage weight in the currently uncovered rows $rf$, ensuring maximal progress per iteration.

**Algorithm 1** CoverageExplanation (CovAXp)

**Input**: $wxp$: sbWCXps matrix, $fcw$: feature weight, $tnr$: desired TNR (def: 1), $ml$: max rule length (def: $|\mathcal{F}|$)
**Output**: *sbAXp*: explanation

```
rf ← {1, ..., |wxp|} {all row indices}
ud ← ∅ {unitary description}
sfeat ← FilterSort(wxp, fcw, rf)
stack ← [(ud, rf, sfeat)]
while stack ≠ [] do
    (ud, rf, sfeat) ← stack.pop()
    for i ∈ sfeat do
        if ¬Valid(wxp, rf, ud, i) then
            continue {i do not contribute}
        end if
        nud_i ← ud ∪ {i ↦ exclusively covered rows}
        nrf ← rf \ covered rows of i
        if TNR(wxp, nrf) ≥ tnr then
            return nud_i {solution found}
        end if
        if |nud_i| < ml then
            next ← FilterSort(wxp, fcw, nrf)
            stack.push((nud_i, nrf, next))
        end if
    end for
end while
return ∅ {no solution found}
```

**Pruning Strategy.** Every candidate feature $i$ is validated by a single function Valid$(wxp, rf, ud, i)$ before it is considered for inclusion. This function combines three necessary conditions to ensure that $i$ contributes to the result guaranteeing that each branch advances toward full coverage while preserving subset-minimality:

- (1) *Coverage progress:* $i$ must cover at least one previously uncovered row in $rf$; that is, there exists a row $r \in rf$ with $wxp[r,i] = 1$ that no already selected feature covers. Otherwise $i$ contributes no progress toward satisfying Condition (1) of Proposition 3.
- (2) *Irreducibility preservation:* after tentatively adding $i$ to the current explanation, every feature in the candidate set must retain at least one row that it alone covers (Condition (2) of Proposition 3). If any previously selected feature loses its unique coverage, the candidate is rejected.
- (3) *Future contribution:* from the previously uncovered rows $rf$, let $j$ be the feature position with the last value 1, if $i > j$ that candidate is rejected. This is a collorary from Condition (1) of Proposition 3, $\forall i \in \{j+1, \ldots, n\}, \nexists i$ that will lead to a final solution, since at least a single row of only zeros will be present in the submatrix. This validation is new with respect to the *RegularSerch* algorithm.

**Algorithm Correctness.** The following proposition establishes necessary and sufficient conditions for identifying a valid *sbAXp*, which form the theoretical foundation for our algorithm's correctness guarantees:

**Proposition 3.** *Let $\mathcal{M}$ be a submatrix of sbWCXp with columns corresponding to the feature set $T \subseteq \mathcal{F}$. $T$ constitutes a sample-based abductive explanation (sbAXp) iff:*

1. $\forall$ *row* $r \in \mathcal{M} : \sum_{j \in T} \mathcal{M}[r, j] \geq 1$ *(full coverage)*
2. $\forall$ *column* $j \in \mathcal{M}$, $\exists$ *row* $r_j \in \mathcal{M}$ *s.t.* $\mathcal{M}[r_j, j] = 1 \wedge (\forall k \in T \setminus \{j\}, \mathcal{M}[r_j, k] = 0)$ *(irreducibility)*

*Proof.* We prove both directions of the equivalence.
**Necessity:** Assume $T$ is a *sbAXp*:

- *Condition (1):* Assume that condition (1) does not hold. Therefore $\exists r^*$ such that $\sum_{j \in T} \mathcal{M}[r^*, j] = 0$. Then $r^*$ corresponds to a $sbWCXp$ not covered by $T$, and therefore $T$ is not a *sbAXp*, which is a contradiction.
- *Condition (2):* Assume that condition (2) does not hold. Therefore $\exists j^* \in T$ such that for every row $r \in \mathcal{M}$ if $\mathcal{M}[r, j^*] = 1$ then $\sum_{k \in T} \mathcal{M}[r, k] \geq 2$. Consider $T' = T \setminus \{j^*\}$, then for every row $r \in \mathcal{M}, \sum_{k \in T'} \mathcal{M}[r, k] > 0$, and therefore $T'$ covers all rows in $\mathcal{M}$ and $T$ is not a *sbAXp*, which is a contradiction.

**Sufficiency:** Assume (1) and (2) hold we will show that $T$ is a *sbAXp*. For this we need to show that:

- Due MHS duality $T$ hits every row in $\mathcal{M}$, Condition (1) directly satisfies $\nexists r, \sum_{j \in T} \mathcal{M}[r, j] = 0$
- For minimality: Assume $T$ is not irreducible. Then $\exists j \in T$ such that $T \setminus \{j\}$ hit all rows. But by (2), $\exists r^*$ such that $\mathcal{M}[r^*, j] = 1$ and $\mathcal{M}[r^*, k] = 0 \; \forall k \neq j$, so $r^*$ is not hit by $T \setminus \{j\}$ which is a contradiction. Thus $T$ is irreducible.

Therefore, $T$ constitutes a valid *sbAXp* if and only if both conditions (1) and (2) hold. □

The following lemma highlights an interesting property derived from Proposition 3 used to show the correctness of the proposed algorithm.

**Lemma 4** ($T$ contains an identity matrix)**.** *For any feature set $T \subseteq \mathcal{F}$ satisfying condition (2) of Proposition 3, the submatrix $\mathcal{M}$ of sbWCXp with the features in $T$, contains a permutation of $|T| \times |T|$ identity matrix as a submatrix.*

CovAXp correctly computes a *sbAXp* as it leverages Proposition 3 as follows. It uses Lemma 4 to add features incrementally to the explanation while preserving Condition (2). In every iteration of the loop, if condition (2) does not hold the candidate feature $i$, then $i$ is not added to the solution and every possible further combination is also ignored as a consequence. When Condition (1) is met, then by Proposition 3 we can ensure that the explanation found is a *sbAXp*.

CovAXp iteratively constructs a *sbAXp* while incorporating two adjustable parameters to balance precision and interpretability. The target TNR ($tnr \leq 1.0$, defaulting to 1) allows controlled relaxation of explanation false positive instances, while the maximum rule length ($ml \leq |\mathcal{F}|$, defaulting to the full feature set) bounds the explanation size. This design enables a TNR-cardinality tradeoff, where the algorithm may terminate before fully satisfying Condition (1) to produce more concise explanations when the current solution meets either the $tnr$ threshold or the $ml$ constraint. When these conditions are unmet, the algorithm pushes a new state onto the stack to explore deeper feature combinations.

**Complexity Analysis.** The algorithm implements an pruned depth-first search in the feature space. Let $m$ denote the number of rows (contrastive instances) in the $wxp$ matrix and $n$ the number of features bounded by $ml$ (default $ml = |\mathcal{F}|$). The algorithm's computational complexity derives from the following components:

- FILTERSORT: Features are sorted by coverage at each step, enabling a greedy prioritization, $\mathcal{O}(m \cdot n)$
- VALID: Evaluates coverage progress, irreducibility and future contribution over at most $m$ rows of the $wxp$ matrix, $\mathcal{O}(m)$

The task of computing a single *sbAXp* is known to be polynomial (Marques-Silva, Lefebre-Lobaina, and Martinez 2025). At each level of the search, the combined cost of FILTERSORT and VALID is $\mathcal{O}(mn + m)$, and the depth is bounded by the explanation length (at most $n$). Consequently, the observed complexity of CovAXp is comparable to that of FastAXp, as confirmed empirically in Section 4.

The computational cost of CovAXp derives from its exploration of the power set of features while leveraging three key optimization strategies. First, the stack depth is strictly bounded by $n$, only reaching the maximum depth when the solution requires all features (equivalent to an identity matrix structure and a trivial case). Second, the contruction of a single sbAXp (rather than all sbAXps) combined with irreducibility and contribution pruning (Condition (2) checked in the main loop) effectively reduces the branching factor to near-constant in practice. At each stack level, only up to $n$ features are evaluated, and entire subspaces are eliminated when a feature cannot satisfy the valid requirements. Third, coverage-based prioritization ensures that features most likely to maximize row coverage are evaluated first, creating a geometric reduction in the remaining search space at each iteration. In practice, the combination of these strategies yields execution times comparable to polynomial-time baselines (Section 4).

**Execution Example.** Consider CovAXp applied to $(\mathbf{v}_9, c_9)$ using the $sbWCXps$ from Table 3, with feature coverage weights $fcw = (3, 2.66, 1.66, 3.66, 4)$ and $tnr = 1$. CovAXp first selects feature $E$ (highest coverage), then $D$, but later $A$ and $B$ violate irreducibility. Finally, feature $C$ contributes to the solution but since no remaining feature left to test the algorithm backtracks to the stage with only feature $E$. The algorithm continue adding features that contribute to the potential soulution starting now with feature $A$. It eventually finds

| Metric | Anc | GrCF | MAXp | FAXp | CAXp |
|---|---|---|---|---|---|
| Max len | 54 | 33 | **6** | 11 | 7 |
| Mean len | 4.16 | 4.54 | **2.36** | 3.44 | 2.48 |
| Std len | 5.77 | 3.59 | **1.12** | 1.77 | 1.29 |
| Max time (s) | 600 | 600 | 215.89 | 2.1 | **1.07** |
| Mean time (s) | 62.0 | 19.6 | 1.36 | 0.15 | **0.11** |
| Std time (s) | 207 | 99.84 | 13.27 | 0.37 | **0.24** |
| Mean TNR | 0.96 | **1.0** | **1.0** | **1.0** | **1.0** |
| Mean TPR | **0.26** | 0.08 | 0.10 | 0.05 | 0.13 |
| Mean acc | **0.50** | 0.43 | 0.44 | 0.41 | 0.46 |
| Mean prec | 0.97 | **1.0** | **1.0** | **1.0** | **1.0** |

Table 5: Aggregate performance comparison.

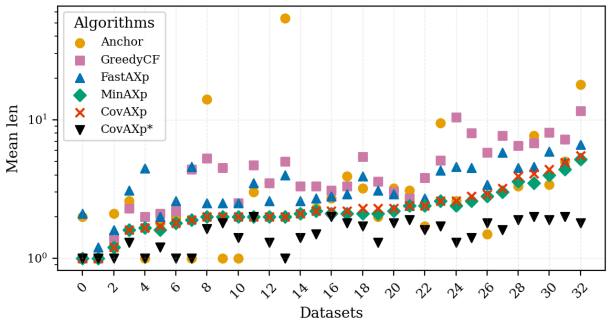


Figure 1: Mean rule length per dataset. CovAXp* denotes relaxed TNR ($tnr = 0.96$).

$R_{Cov} = (E = 1 \wedge A = 1 \wedge B = 1 \wedge C = 1) \rightarrow 1$ after exploring the alternative branch. This example illustrates that CovAXp does not always find the cardinality-minimal explanation, yet correctness and formal guarantees are preserved. The initial state uses $ud = \emptyset$ and $rf$ containing all rows of $wxp$.

The next section demonstrates this empirically, showing how CovAXp achieves: (1) sub-second execution times on high-dimensional datasets, (2) near-linear scaling in rule length discovery. The observed performance aligns with our theoretical analysis, confirming that coverage prioritization and irreducibility pruning effectively transform an exponential search problem into a polynomial-time solution.

## 4 Experimental Evaluation

This section evaluates CovAXp against baseline approaches. In the experiments, we evaluated the results through different evaluation metrics, focusing on the cardinality of obtained rules and the computational efficiency.

### 4.1 Experimental Setup

We evaluated 35 real-world classification datasets from the UCI ML repository and OpenML[2] spanning Boolean, categorical, integer, and real-valued features, with dimensionality ranging from 5 to 1,617 features and from 116 to 204703 instances[3]. All datasets are concerned with solving supervised classification problems with 2 or more classes. For each dataset, we explained 10 different instances randomly selected (350 in total), with reported metrics representing means across all instances, similar to a 10-fold cross-validation (Refaeilzadeh, Tang, and Liu 2009). All experiments were carried out on an Asus ROG Zephyrus G14 with an AMD Ryzen™ 7 4800HS processor (2.9 GHz) and 16 GB RAM.

We compared four approaches, selected from the model-agnostic explainability literature with an emphasis on feature-selection methods that produce logical explanations. Anchors was included due to its prominence in the XAI community, despite lacking the same formal guarantees as the other approaches. MinAXp provides the theoretical optimum for explanation length, FastAXp provides the theoretical baseline for polynomial-time complexity, and GreedyCF represents a heuristic middle ground. The algorithm parameters were configured as follows:

- **Anchors** (Anc): The model was set to default values except for the sampling size (10k)
- **GreedyCF** (GrCF): The sub-matrix size for searching all *sbAXps* was defined as $|k| = 10$
- **MinAXp** (MAXp): Optimal minimal explanation (theoretical length baseline using a SAT solver)
- **FastAXp** (FAXp): Polynomial time explanation (theoretical complexity baseline)
- **CovAXp** (CAXp): Our proposed algorithm with default configuration ($tnr = 1.0$ and $ml = |\mathcal{F}|$)

To make a more fair comparison, since Anchors sampling is parameterized, for all algorithms we selected the number of rows in the training dataset up to 10,000 instances as the sampling evaluation ($\mathbf{S}$). For every experiment, the similarity function per feature (*sf*) used to compare the instances was the strict equality. All datasets were preprocessed using the same discretization strategy used by Anchors and the model trained to explain was a Random Forest classifier.

### 4.2 Performance Analysis

We evaluated the algorithms using four key metrics: cardinality (len), which quantifies explanation size; execution time (in seconds) with a 600-second timeout threshold; accuracy (acc), precision (prec), TNR and TPR of the explanation within the sampling. The latter metrics are defined in Table 1. Some of the evaluation metrics related to explainability are also included in the measures proposed.

- **Fidelity**: Is measured trough the accuracy with respect to the sampling $\mathbf{S}$
- **Comprehensibility**: Comprehensibility is also included by measuring the rule length. Other aspects like number of rules, redundancy and consistency are pointless since all evaluated algorithms provide a single rule as output.
- **Stability**: In (Marques-Silva, Lefebre-Lobaina, and Martinez 2025) is explained how *sbWCXps* are pessimistic and won't change when changing $\mathbf{S}$ and how *sbAXps* are

[2]https://archive.ics.uci.edu/ and https://www.openml.org/
[3]Supplementary material have all datasets and sources

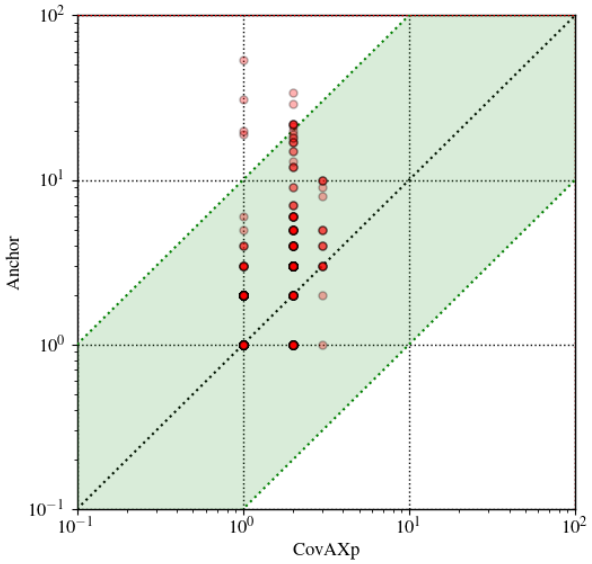


Figure 2: Length comparison (log scale) between CovAXp and Anchors across all instances.

optimistic and may change only when a new *sbCXp* appears. Since all evaluated methods (except Anchors) provide sample-based rigorous explanations and we can predict the required perturbations to change the predictions we don't measure it in this work.

Of 350 evaluated instances in total, only 322 were used in the evaluation due timeouts/memouts for Anchors and GreedyCF. Table 5 summarizes the comparative performance in all datasets and instances, highlighting in bold the best performance for each metric. The minimum time was excluded from the table because it has sub-second values for all the algorithms, also the minimum length, maximum precision and maximum TNR was excluded because was 1 for every algorithm compared. From the results obtained, we have several key observations:

1. **Efficiency**: CovAXp reduced the mean execution time versus other approaches, the maximum runtime showed even greater improvements.
2. **Explanation Quality**: CovAXp maintained near-minimal cardinality, with a mean of 2.48 compared to MinAXp's 2.36. It achieves a TNR of 1.0, matching MinAXp, FastAXp and GreedyCF. In contrast, Anchor failed to meet perfect TNR in 256 instances, and for 80 of those cases, it's TNR fell below 0.96. Regarding accuracy, CovAXp reached a score of 0.46, which was competitive with Anchors 0.50, while also ensuring formal guarantees.
3. **Robustness**: CovAXp exhibited no timeouts and sub-second execution in all cases, while Anchors and GreedyCF timed out (600s) on several instances.

## 4.3 Rule Length Analysis

We evaluated in more details the explanation size of each algorithm output for 34 datasets. One dataset was excluded due to timeouts from Anchors across all instances. Figure 1 shows that CovAXp demonstrates strong alignment with theoretically minimal explanations, only providing larger explanations in 36 of 350 evaluated instances (10.3% of instances). Although Anchors produced shorter rules than CovAXp in 10 datasets (29.4%), also in all of these cases the rules were also shorter than MinAXp, meaning that the rules from Anchors will not cover all contrastive cases, in other words this advantage is achieved with TNR $< 1$ on several instances. Also, Anchors have 28 incidence of timeouts/memouts across instances (several of them in datasets like Residential Building [4]), and mean cardinality 1.6 times larger than CovAXp's. In the Figure 2 we present a instance level comparison between Anchors and CovAXp (darker points means more instances showing that difference) to show more clearly the difference.

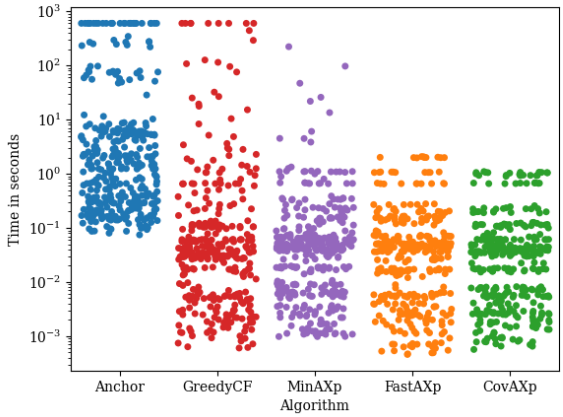


Figure 3: Execution times (log scale) per instance.

The remaining algorithms have equal or higher rule length than CovAXp in all evaluated instances. The TNR-relaxed CovAXp* variant was configured at TNR $= 0.96$ to match Anchors mean TNR and demonstrated superior cardinality reduction. The key advantage of CovAXp* over CovAXp is that it enables explicit control of the TNR–cardinality trade-off: by relaxing the TNR threshold, the user can obtain shorter rules with guaranteed minimum TNR, a flexibility not offered by the other baselines. This variant outperformed Anchors in explanation length in 30 of 34 datasets and reduced cardinality versus the theoretical optimum (MinAXp) in 28 datasets. This tunability represents a significant advantage for applications where near-perfect TNR is acceptable. Finally, GreedyCF and FastAXp consistently generated equal or longer explanations than CovAXp across all datasets.

## 4.4 Computational Efficiency

We evaluated running times across all datasets and instances. Figure 3 demonstrates CovAXp's significant computational

[4]UCI ML repository https://archive.ics.uci.edu/dataset/437/residential+building+data+set

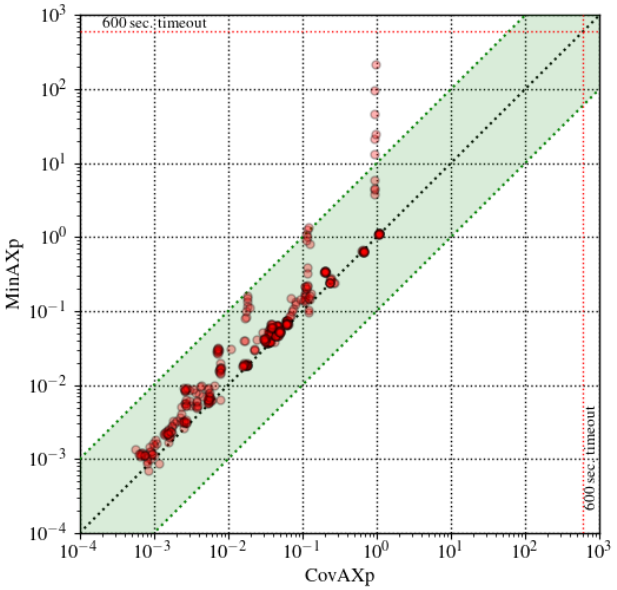


Figure 4: Execution times (log scale) per instance between CovAXp and MinAXp.

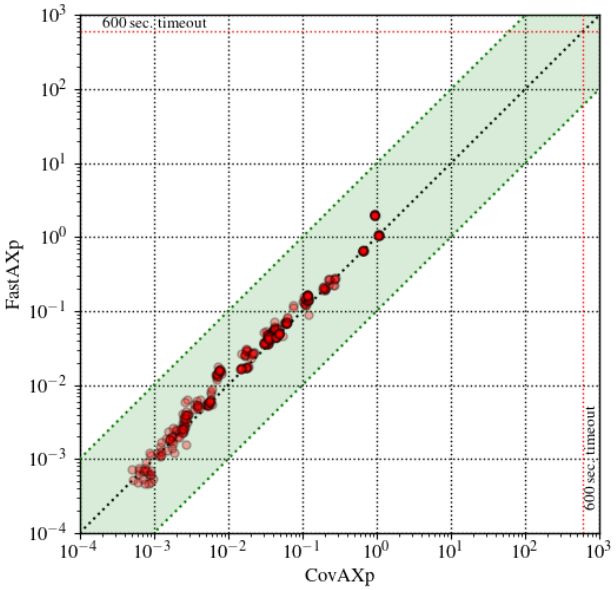


Figure 5: Execution times (log scale) per instance between CovAXp and FastAXp .

advantages: it maintained stable sub-second performance across all instances while achieving speedups over MinAXp on complex datasets. Notably, CovAXp eliminated the timeouts experienced by both Anchors and GreedyCF approaches and exhibited minimal runtime variance compared to alternative methods.

Figures 4 and 5 detail instance-level time comparisons. CovAXp's pruning strategy yields faster execution for most instances; even where other algorithms outperform it, the margin is small and CovAXp still produces shorter explanations. These results show CovAXp's suitability for real-world deployment, where both computational efficiency and explanation quality are critical.

## 5 Conclusion

This paper introduces CovAXp, a novel algorithm that bridges the critical gap between formal guarantees and practical efficiency in explainable AI. Subset and cardinality-minimal explanations provide formal and explanation size guarantees but at high computational cost. Other approaches that also provide formal guarantees are more efficient in running time but cannot ensure a small explanation size. Using feature coverage heuristics within a sample-based logic framework, CovAXp achieves rigorous abductive explanations while overcoming the computational limitations of traditional logic-based methods. Our approach fundamentally addresses two key challenges in XAI: 1) the TNR-reliability tradeoffs in model-agnostic approaches, and 2) the tradeoff between explanation minimality and computational tractability.

The experimental evaluation across 35 real-world datasets demonstrates that CovAXp achieves:

- **Near-optimal minimality** with only 10% larger explanations on average than the theoretical optimum, while maintaining TNR value of 1.
- **Competitive efficiency**, preserves polynomial complexity running time.
- **Tunable TNR-cardinality tradeoffs** relaxing TNR allows to obtain smaller explanations with minimal TNR compromise, or target a desired length.

These advancements establish a new practicality threshold for rigorous XAI. Unlike other sampling-based methods, CovAXp maintains the mathematical rigor of abductive explanations while achieving computational performance suitable for deployed systems of XAI. The algorithm design, particularly its feature coverage heuristic and row filtering mechanism, provides a blueprint for balancing theoretical soundness with empirical efficiency.

Future work will explore four promising directions: 1) definition of a testing framework for building synthetic datasets to evaluate and compare explanation algorithms in different scenarios; 2) definition of sampling strategies to improve the quality of provided *sbAXps*; 3) extension to similarity functions that are not restricted to binary output; and 4) integration with emerging regulatory frameworks for certified AI explanations. The CovAXp approach opens new possibilities for deploying trustworthy AI in high-stakes domains where both auditability and performance are mandatory requirements.

## AI Declaration

The authors used ChatGPT by OpenAI exclusively to improve grammar, spelling, and readability of the manuscript. The authors reviewed and approved all edits and take full responsibility for the content.

## Acknowledgements

The authors also acknowledge support by the Spanish MCIN/AEI (CHIST-ERA iTrust) project PCI2022-135010-2, project PID2022-139835NB-C21, PIE 2023-5AT010, and VALAWAI, Spain (Horizon Europe#101070930). Also by the projects PID 2023-152814OB-I00, HE-101070930, and by ICREA starting funds.

## References

Alangari, N.; El Bachir Menai, M.; Mathkour, H.; and Almosallam, I. 2023. Exploring evaluation methods for interpretable machine learning: A survey. *Information* 14(8):469.

Alvarado-Moreira, R., and Ibarra-Fiallo, J. 2024. New approach to facial expression recognition and classification using typical testors. In *Intelligent Systems Conference*, 406–414. Springer.

Amgoud, L.; Cooper, M. C.; and Debbaoui, S. 2024. Axiomatic characterisations of sample-based explainers. In *ECAI*, 770–777.

Bastani, O.; Kim, C.; and Bastani, H. 2017. Interpreting blackbox models via model extraction. *arXiv preprint arXiv:1705.08504*.

Chamola, V.; Hassija, V.; Sulthana, A. R.; Ghosh, D.; Dhingra, D.; and Sikdar, B. 2023. A review of trustworthy and explainable artificial intelligence (xai). *IEEe Access* 11:78994–79015.

Cooper, M., and Amgoud, L. 2023. Abductive explanations of classifiers under constraints: Complexity and properties. In *ECAI 2023*. IOS Press. 469–476.

Crama, Y.; Hammer, P. L.; and Ibaraki, T. 1988. Cause-effect relationships and partially defined boolean functions. *Annals of Operations Research* 16(1):299–325.

Darwiche, A. 2023. Logic for explainable AI. In *LICS*, 1–11.

Fawcett, T. 2006. An introduction to roc analysis. *Pattern recognition letters* 27(8):861–874.

Fredman, M. L., and Khachiyan, L. 1996. On the complexity of dualization of monotone disjunctive normal forms. *Journal of algorithms* 21(3):618–628.

Fürnkranz, J.; Gamberger, D.; and Lavrač, N. 2012. *Foundations of rule learning*. Springer Science & Business Media.

Gainer-Dewar, A., and Vera-Licona, P. 2017. The minimal hitting set generation problem: algorithms and computation. *SIAM Journal on Discrete Mathematics* 31(1):63–100.

Geng, Z.; Schleich, M.; and Suciu, D. 2022. Computing rule-based explanations by leveraging counterfactuals. *Proc. VLDB Endow.* 16(3):420–432.

Han, J.; Pei, J.; and Tong, H. 2022. *Data mining: concepts and techniques*. Morgan kaufmann.

Huysmans, J.; Dejaeger, K.; Mues, C.; Vanthienen, J.; and Baesens, B. 2011. An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. *Decision Support Systems* 51(1):141–154.

Ignatiev, A.; Narodytska, N.; Asher, N.; and Marques-Silva, J. 2020. From contrastive to abductive explanations and back again. In *AIxIA*, 335–355.

Ignatiev, A. 2020. Towards trustable explainable AI. In *IJCAI*, 5154–5158.

Karimi, M. 2025. *Stability of Post-hoc Explanations under Different Training Objectives*. Ph.D. Dissertation, Hochschule Mittweida.

Lakkaraju, H.; Bach, S. H.; and Leskovec, J. 2016. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1675–1684.

Lazo-Cortes, M.; Ruiz-Shulcloper, J.; and Alba-Cabrera, E. 2001. An overview of the evolution of the concept of testor. *Pattern recognition* 34(4):753–762.

Lefebre-Lobaina, J. A., and Shulcloper, J. R. 2023. Regularsearch, a fast performance algorithm for typical testors computation. *Information Sciences* 649:119665.

Lundberg, S. M., and Lee, S. 2017. A unified approach to interpreting model predictions. In *NeurIPS*, 4765–4774.

Marques-Silva, J., and Ignatiev, A. 2022. Delivering trustworthy AI through formal XAI. In *AAAI*, 12342–12350.

Marques-Silva, J.; Lefebre-Lobaina, J.; and Martinez, V. 2025. Efficient and rigorous model-agnostic explanations. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, 2637–2646.

Mohammed, J. Z., and Wagner, M. 2014. Data mining and analysis: fundamental concepts and algorithms. *Cambridge University*.

Pons-Porrata, A.; Gil-García, R.; and Berlanga-Llavori, R. 2007. Using typical testors for feature selection in text categorization. In *Iberoamerican Congress on Pattern Recognition*, 643–652. Springer.

Refaeilzadeh, P.; Tang, L.; and Liu, H. 2009. Cross-validation. In *Encyclopedia of database systems*. Springer. 532–538.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "why should I trust you?": Explaining the predictions of any classifier. In *KDD*, 1135–1144.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2018. Anchors: High-precision model-agnostic explanations. In *AAAI*, 1527–1535.

Rudin, C., and Shaposhnik, Y. 2023. Globally-consistent rule-based summary-explanations for machine learning models: application to credit-risk evaluation. *Journal of Machine Learning Research* 24(16):1–44.

Wang, K.; He, Y.; and Han, J. 2003. Pushing support constraints into association rules mining. *IEEE Transactions on Knowledge and Data Engineering* 15(3):642–658.

Yang, F.; He, K.; Yang, L.; Du, H.; Yang, J.; Yang, B.; and Sun, L. 2021. Learning interpretable decision rule sets: A submodular optimization approach. *Advances in Neural Information Processing Systems* 34:27890–27902.

Yang, F.; Yang, Z.; and Cohen, W. W. 2017. Differentiable learning of logical rules for knowledge base reasoning. *Advances in neural information processing systems* 30.

Yang, L.; Yang, J.; and Sun, L. 2024. Efficient decision rule list learning via unified sequence submodular optimization. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 3758–3769.

Yin, X., and Han, J. 2003. Cpar: Classification based on predictive association rules. In *Proceedings of the 2003 SIAM international conference on data mining*, 331–335. SIAM.

Zaki, M. J. 2000. Generating non-redundant association rules. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, 34–43.