Comparative Analysis of Surrogate Model Architectures for LIME f or Intrusion Detection Enhancement*

Mantas Bacevicius[1,*,†], Agne Paulauskaite-Taraseviciene[1,2,†]

[1]Kaunas Technology University, Faculty of Informatics, Studentu 50, Kaunas, 51368, Lithuania
[2]Centre of Excellence for Sustainable Living and Working (SustAInLivWork)

Abstract

Machine learning has proven highly effective for network intrusion detection but remains opaque in its decision-making, creating trust and interpretability concerns in cybersecurity. In this work, we refine Local Interpretable Model-Agnostic Explanations (LIME) by examining how different surrogate regressors (Decision Tree, Random Forest, Ridge) and perturbation distributions (Beta, Gamma, Gaussian, Pareto, Weibull) affect explanation quality for the CIC-IDS-2018 dataset. Our experiments on four classifiers (Decision Tree, kNN, Random Forest, XGBoost) reveal that tree-based surrogates, especially Decision Tree and Random Forest, can achieve near-perfect or even perfect fidelity ($R^2 \approx 1.0$) under distributions like Beta, Gamma, and Pareto, significantly outperforming the default linear Ridge regression surrogate. These distributions also improve the stability of the explanation, as measured by the Jaccard similarity and cosine similarity of the highest scoring traits, and the Pareto and Beta distributions often provide the highest consistency across repeated tests. In summary, our results highlight the importance of balancing the complexity of the surrogate model with the non-linearity of the target classifier and selecting appropriate perturbation distributions to increase both accuracy and stability. Hence, this work proposes a systematic framework to improve interpreted intrusion detection, allowing cybersecurity applications to provide more robust and adaptable local explanations.

Keywords

Intrusion Detection Systems, Explainable AI, LIME, Stability, Fidelity

1. Introduction

Modern machine learning models (e.g., ensemble methods and deep neural networks) often operate as black boxes, making it difficult to understand why a particular decision was made. This lack of transparency is problematic in high-stakes domains like cybersecurity, where understanding why an intrusion is detected is as important as detecting it. To address this, post-hoc explanation techniques have been developed to illuminate model behavior without altering the underlying model. A prominent example is Local Interpretable Model-Agnostic Explanations (LIME), which explains an individual prediction by learning a simple surrogate model in the vicinity of that prediction [1]. In LIME, one perturbs the input around the instance of interest, observes the black-box model's outputs, and then fits an interpretable model (originally a sparse linear regressor) to mimic those local outputs [2]. The surrogate is trained on perturbed samples weighted by their similarity to the target instance, ensuring the surrogate focuses on the local decision boundary. If successful, the surrogate achieves high local fidelity, meaning it approximates the black-box model's behavior well in that local region [2]. This allows LIME to provide interpretable explanations (e.g. feature importance weights) that are locally faithful to the complex classifier's prediction. While LIME has proven useful across domains, two key limitations hinder its reliability: instability and imperfect fidelity. Instability refers to the sensitivity of LIME's explanations to the randomness in its sampling procedure – repeating LIME on the same instance can yield different explanations [3]. Ribeiro et al. acknowledged the need for faithful explanations, but in practice the random perturbations can lead to inconsistent feature weights for the same prediction

*IVUS 2025: Information Society and University Studies 2025, May 15, Kaunas, Lithuania
*Corresponding author.
†These authors contributed equally.
mantas.bacevicius@ktu.edu (M. Bacevicius); agne-paulauskaite-taraseviciene@ktu.lt (A. Paulauskaite-Taraseviciene)
0009-0007-6772-8898 (M. Bacevicius); 0000-0002-8787-3343 (A. Paulauskaite-Taraseviciene)

[4]. This variance undermines user trust, since an explanation should ideally be robust for a given input. Meanwhile, fidelity issues arise because LIME's default surrogate (a regularized linear model) may be too simple to capture complex local behavior. If the true model's decision boundary is highly non-linear or involves feature interactions, a linear surrogate might fit a poor local approximation, reducing the explanation's faithfulness to the original model. Prior studies have noted these problems: LIME "suffers from critical problems such as instability and local fidelity" that can limit its usefulness in real-world settings [5]. As a result, several improvements to LIME have been proposed. For example, Bootstrap-LIME (B-LIME) uses bootstrapping and modified sampling to produce more stable and locally faithful explanations. Other variants introduce regularization or adjust the locality parameters to balance fidelity and interpretability. These efforts report more consistent explanations and better local alignment with the original model, confirming that there is room to improve LIME's stability and fidelity without sacrificing interpretability. In this paper, we investigate a surrogate model refinement to enhance LIME's explanations. In the original LIME, a Ridge (L2-regularized) linear regressor is used as the surrogate explainer. We propose to replace this with tree-based regression models – specifically, a `Decision Tree` and a `Random Forest` regressors – as the local surrogate. The motivation is that tree-based models can capture non-linear relationships and interaction effects in the neighborhood of an instance better than a linear model, potentially improving local fidelity of the explanation. A Decision Tree surrogate can partition the local feature space and may align more naturally with the behavior of tree-based classifiers (like Decision Tree ensembles), while an ensemble of trees (Random Forest) can further smooth out irregularities, reducing variance in the surrogate fit. Notably, Decision trees are themselves interpretable (providing rule-based explanations), and even a Random Forest can be interpreted via feature importance or by examining constituent trees. Thus, these surrogates offer a richer hypothesis space for local explanation without completely forfeiting interpretability. We hypothesize that these LIME modifications will yield explanations that are more faithful to complex models and more stable against sampling variations. We evaluate the modified LIME approaches on a network intrusion detection case study using the CIC-IDS-2018 dataset. We train four different classification models – a single Decision Tree, a Random Forest ensemble, a k-Nearest Neighbors (kNN) classifier, and an XGBoost (Gradient-Boosted tree) model – to distinguish between benign traffic and various attack types in the dataset. We then generate explanations for individual predictions using three explainer variants: the original LIME (with Ridge regression), LIME with a Decision Tree surrogate, and LIME with a Random Forest surrogate. Experimental results show that our tree-based surrogates significantly improve the quality of explanations on both metrics of interest: (1) Stability – explanations (feature importance rankings) remain more consistent when LIME is repeated or perturbed, compared to the high variance observed with linear surrogates [3]; and (2) Local Fidelity – the surrogate's predictions match the black-box model's outputs in the neighborhood of the instance more closely (higher local R^2 and lower approximation error), indicating the explanation truly reflects the model's local behavior. These improvements are observed across all four classifier types, with particularly strong gains for the more complex models (Random Forest and XGBoost) where a linear explanation struggles to capture non-linear decision boundaries.

2. Related Works

LIME (Local Interpretable Model-Agnostic Explanations) is a popular technique that explains an individual prediction by training a simple surrogate model (e.g. a linear regressor) on synthetic data points generated around the instance. However, LIME's reliance on random perturbations causes high variance in the selected features, and its linear surrogate can struggle to faithfully mimic the original model's local behavior [6, 7]. In fact, a theoretical analysis showed that poor parameter choices can even cause LIME to miss important predictive features [8]. These stability and fidelity issues have been well documented, motivating the development of numerous LIME extensions. Researchers have even proposed quantitative metrics to gauge explanation reliability (stability) [9], underlining the need for more consistent and faithful explanations in critical applications. One line of research tackles

LIME's instability by removing or reducing its randomness. Zafar and Khan's DLIME [6] replaces LIME's random sampling with a deterministic sampling strategy: it clusters the training data and uses the nearest cluster for generating perturbations, yielding much more consistent explanations. This approach substantially improves explanation overlap across runs (e.g. higher Jaccard similarity), though using a limited cluster can sometimes reduce local accuracy. Zhou et al. [10] introduced S-LIME, which employs a hypothesis-testing framework to adaptively increase the number of samples until the explanation stabilizes – effectively guaranteeing a desired level of consistency at the cost of additional computation. Other works inject probabilistic structure: Zhao et al. [11] proposed BayLIME, a Bayesian reformulation of LIME that introduces prior knowledge into the surrogate model, achieving more stable feature weights across runs and improving robustness to kernel parameters. Visani et al. [12] took an optimization approach with OptiLIME, tuning LIME's kernel width and regularization to find an optimal balance between explanation variance and fidelity. Recently, Bora et al. [13] addressed LIME's inconsistency in image tasks with SLICE, which refines superpixel selection and uses adaptive perturbations to produce significantly more stable (and even more faithful) explanations for vision models. Domain-specific variants have also emerged; for example, Abdullah et al. [5] developed B-LIME for time-series data, applying bootstrap resampling and sequence-aware perturbations to bolster explanation stability in ECG signal classification. Another line of enhancements focuses on improving LIME's local fidelity – how well the surrogate approximates the true model in the vicinity of the instance. Shankaranarayana and Runje's ALIME [14] employs an autoencoder to generate perturbations on the data manifold and to weight samples by their reconstruction error, yielding more plausible neighborhood examples; this modification was shown to produce explanations that are both more stable and more faithful to the model's predictions than standard LIME. Similarly, Shi et al. [15] introduced a method to respect feature dependencies in the perturbation process and even fit a nonlinear local model to capture complex decision boundaries, which significantly boosted the surrogate's fidelity to the black-box model. More recently, Tan et al. [7] proposed GLIME, a generalization of LIME that draws truly local, unbiased sample points by integrating the locality weighting into the sampling distribution. GLIME achieves markedly higher surrogate R^2 (local fidelity) and nearly deterministic explanation results, dramatically outperforming vanilla LIME on both stability and fidelity metrics. In a related vein, Cinquini and Guidotti [16] developed CA-LIME (Causality-Aware LIME), which replaces LIME's naive random sampling with a causally-informed data generator to ensure only realistic perturbations. By encoding causal relationships among features, CA-LIME attains better agreement with the original model's behavior, improving both the consistency of explanations and their fidelity to the true model.

3. Dataset

The CIC-IDS-2018 dataset, a collaborative effort between the Canadian Institute for Cybersecurity (CIC) and the Communications Security Establishment (CSE) [17], is a significant resource for network intrusion detection research. It comprises network traffic and system logs from a simulated environment, capturing both benign and malicious activities over ten days, totaling approximately 16,233,002 instances. This section provides a comprehensive analysis of its applicability with Random Forest, Decision Tree, kNN, and XGBoost machine learning models, alongside the use of LIME for explainable AI (XAI), and details the main disadvantages of the dataset. The dataset includes multiple different attack scenarios including: Brute-force, Heartbleed, Botnet, DoS, DDoS, Web attacks, and infiltration of the network from inside. The attacking infrastructure includes 50 machines, and the victim organization has 5 departments with 420 PCs and 30 servers. It is distributed over ten CSV files, with nine files consisting of 79 independent features and one file with 83 features, extracted using CICFlowMeter-V3, resulting in 80 features in total [18].

3.1. Applicability with Machine Learning Models

The dataset's structure makes it suitable for various machine learning approaches, particularly for anomaly-based intrusion detection. Table 1 summarizes key performance metrics from the literature,

Table 1
ML model accuracy with CIC-IDS-2018 dataset.

Author(s)	Model(s)	Acc. (%)	Prec. (%)	Recall (%)
D'hooge et al. [20]	DT, RF, XGB, kNN	96	99	79
Huancayo Ramos et al [21].	DT, RF	99	100	99
Karatas et al. [22]	k-NN, RF, GB	99	99	99
Fitni et al. [23]	GB, LR, DT	98	98	97

based on a survey by the Journal of Big Data [19].

Random Forest (RF) and Decision Tree (DT) have shown strong performance, with Huancayo Ramos et al. [21] reporting Random Forest and Decision Tree achieving 99.99 % accuracy, with 100 % precision and 99.99% recall, indicating their effectiveness in classifying traffic. D'hooge et al. [20] also used Random Forest, achieving 96% accuracy, 99% precision, and 79% recall, suggesting robustness but highlighting potential challenges with recall for minority classes. k-Nearest Neighbors (kNN) has been applied in studies like Karatas et al. [22], where it was part of a suite of models, though Adaboost outperformed it with 99.69% accuracy. XGBoost model has demonstrated high efficacy, with Fitni et al. [23] resulting in 98.80% accuracy in an ensemble including DT, and D'hooge et al. [20] showing strong results alongside other models. Its ability to handle large datasets and class imbalance makes it particularly suitable for CIC-IDS-2018.

These models often perform well, with accuracies often above 95%, but the survey indicates that these results may lead to overfitting, especially given the imbalance in the classes of the dataset.

3.2. Integration with LIME XAI Methodology

LIME, a model-agnostic explainability tool, can be integrated with Random Forest, Decision Tree, kNN, and XGBoost to provide local explanations for predictions. While direct studies using LIME with CIC-IDS-2018 are limited, its applicability is theoretically sound given its compatibility with any black-box model. This is crucial for intrusion detection, where understanding why a model flags certain traffic as an attack can enhance trust and usability in security operations.

3.3. Problematic Aspects

Despite its utility, the CIC-IDS-2018 dataset presents several challenges that researchers and practitioners must address. First of all the dataset exhibits a significant class imbalance as it shown in Table 2 with approximately 17% of instances being attack traffic. This imbalance can bias models toward the majority class (normal traffic), potentially leading to poor recall for attack detection. For example, D'hooge et al. [19] reported 79% recall, suggesting that 21% of attacks might be missed, which is critical in security contexts. Moreover, it is usually a big data challenge, as the dataset's size (over 16 million instances) poses significant computational difficulties. Processing such volumes requires substantial memory and processing power, which may limit accessibility for researchers with limited resources. Detailed analysis reveals specific mislabeling problems, particularly with certain attack types. For instance, the "Error prevalence in NIDS datasets" [24] documentation highlights issues with FTP Patator, where attacks on port 21 were mislaunched as DoS Slowhttptest, and SSH Patator, with 30 flows mislabeled due to incorrect port usage. DoS GoldenEye has about 25% of flows incorrectly labeled as malicious due to short durations, and Web Attack - XSS shows 41.7% mislabeling on certain days. These errors can compromise model training, leading to unreliable predictions in real-world scenarios.

4. Methods and metrics

Excluding rare or anomalous classes can lead to a more balanced, cleaner dataset, thereby boosting IDS performance. These datasets serve as a strong foundation for furthering IDS research, enabling the

Table 2
CIC-IDS-2018 class distribution.

Class	Entry count	Percentage
Benign	13 484 708	83.0700
DDOS attack-HOIC	686 012	4.2260
DDoS attacks-LOIC-HTTP	576 191	3.5495
DoS attacks-Hulk	461 912	2.8455
Bot	286191	1.7630
FTP-BruteForce	193 360	1.1912
SSH-Bruteforce	187 589	1.1556
Infiltration	161934	0.9976
DoS attacks-SlowHTTPTest	139 890	0.8618
DoS attacks-GoldenEye	41 508	0.2557
DoS attacks-Slowloris	10990	0.0677
DDOS attack-LOIC-UDP	1730	0.0107
Brute Force-Web	611	0.0038
Brute Force-XSS	230	0.0014
SQL Injection	87	0.0005

Table 3
Multi-class classification model accuracy results.

Model	Accuracy (weighted)	Precision (weighted)	Recall (weighted)	F1 (weighted)
RF	0.99875	0.98262	0.98640	0.98311
DT	0.99724	0.98072	0.98505	0.98222
KNN	0.99583	0.98021	0.98217	0.97949
XGB	0.99954	0.98343	0.98668	0.98264

creation of models capable of effectively detecting both traditional and modern attack vectors. Therefore, our dataset, based on CIC-IDS-2018 was reduced by removing features which did not correlate with the label or highly correlated with other features. Resulting dataset comprised of 28 classes. Moreover, after filtering, 32 features remained: 7 binary categorical features (with skewed frequency) and 25 numerical features.

This reduced dataset was used to train KNN, RF, DT and XGBoost models to evaluate LIME methodology's explanation stability and fidelity. Models and their training parameters were based on the research by Bacevicius et al. [25] XGBoost (XGB), optimized with $n\_estimators = 200$, $max\_depth = 6$, and a $learning\_rate = 0.1$, achieved the highest weighted accuracy of 0.9995, complemented by a macro-level precision of 0.93256 and recall of 0.844. RF utilizing $n\_estimators = 100$ and $max\_depth = 30$, secured the second-highest weighted accuracy at 0.99875. DT with $max\_depth = 30$ and kNN set at $n\_neighbors = 3$ also performed effectively, their macro-level metrics were slightly lower in comparison. Table 3 highlights strong overall performance across all models, validated with 5-fold cross-validation.

4.1. Stability (Jaccard Similarity)

Stability in the context of explanations refers to the consistency of an explanation under slight variations – ideally, repeated runs of an explanation method (or small perturbations of the input or model) should yield similar explanations. This property is crucial for trust in explanation methods: if an identical prediction can be explained in wildly different ways, it undermines confidence in the explanations. To quantify stability, a common approach is to measure the overlap between sets of features (or factors) identified as important in multiple explanations. A widely used metric for this is the Jaccard similarity coefficient, which compares two sets by the ratio of their intersection size to their union size. Given

two sets A and B (e.g. the sets of top features from two explanation runs), the Jaccard similarity is defined as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

where $|A \cap B|$ is the number of elements common to both sets and $|A \cup B|$ is the total number of unique elements in either set.
This yields a value in $[0, 1]$: $J = 1$ indicates the two explanations select an identical set of features, while $J = 0$ indicates no overlap at all. A higher Jaccard score thus signifies greater stability (more consistent explanations). In theory, a perfectly stable explanation method would produce the same feature set every time (maximizing overlap).

4.2. Fidelity (R^2)

Fidelity refers to how well an explanation model or explanation reflects the original model's behavior. In other words, fidelity measures the accuracy of the explanation in approximating the predictions of the true model. A common way to quantify fidelity for explanation methods like LIME (which produce a surrogate model) is to use the coefficient of determination (R^2) between the surrogate's predictions and the original model's predictions. This R^2 score (also known as the goodness-of-fit) captures the proportion of variance in the model's outputs that is explained by the surrogate explanation model. Formally, if $f(x)$ denotes the black-box model's output for input x and $g(x)$ denotes the surrogate (explanation) model's output, one can collect a set of N samples (typically the perturbed instances in LIME's local neighborhood) and compute R^2 as:

$$R^2 = 1 - \frac{\sum_{i=1}^{N} \big(f(x_i) - g(x_i)\big)^2}{\sum_{i=1}^{N} \big(f(x_i) - \overline{f(X)}\big)^2} \tag{2}$$

where $\overline{f(X)}$ is the average value of the model's output $f(x)$ over the sampled points. The numerator $\sum(f(x_i) - g(x_i))^2$ is the residual sum of squares (the discrepancy between the model and explanation), and the denominator $\sum(f(x_i) - \overline{f(X)})^2$ is the total sum of squares (variance of the model's output in that region).
An R^2 of 1 indicates perfect fidelity – the explanation model g exactly reproduces the black-box f on those samples. An R^2 of 0 means the surrogate is no better than a constant prediction (i.e. it explains 0% of the output variance), and negative values indicate even worse fit (the surrogate deviates more from f than a constant would). In explanation terms, high fidelity means the explanation is faithful to what the model is actually doing, whereas low fidelity means the explanation is potentially misleading or incomplete in describing the model's behavior.

4.3. Cosine Similarity

Cosine similarity quantifies the directional agreement between the predictions of the original (black-box) model and the surrogate (explanation) model. Rather than assessing the amount of variance explained—as in the R^2 metric—cosine similarity focuses on how consistently the surrogate captures the trend in the model's outputs across different input instances. Formally, let $f(x)$ denote the output of the black-box model for an input x, and let $g(x)$ denote the output of the surrogate explanation model. For a set of N samples (typically the perturbed instances in the local neighborhood), define the prediction vectors:

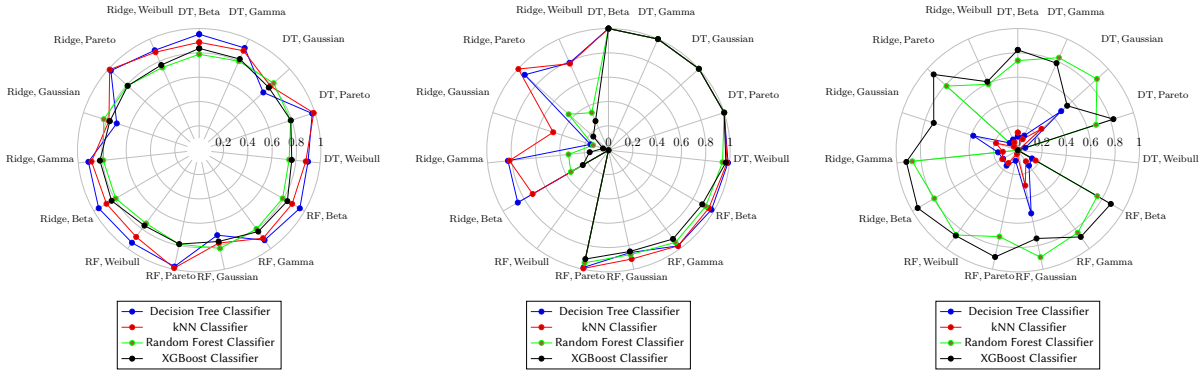
$$\mathbf{f} = \big(f(x_1), f(x_2), \ldots, f(x_N)\big),$$
$$\mathbf{g} = \big(g(x_1), g(x_2), \ldots, g(x_N)\big).$$

The cosine similarity between these vectors is given by:

$$\text{CS} = \frac{\sum_{i=1}^{N} f(x_i)\, g(x_i)}{\sqrt{\sum_{i=1}^{N} \big(f(x_i)\big)^2}\, \sqrt{\sum_{i=1}^{N} \big(g(x_i)\big)^2}}. \tag{3}$$

Table 4
Variables used for experiments.

Surrogate models	Classification models	Test instances for explanations
For local explanations these models were trained with local pertubations and used for explanation generation: • Ridge Regressor; • DT Regressor; • RFRegressor;	Four models with varying levels of complexity were used: • DT; • RF; • kNN; • XGBoost;	Explanations were generated from one random test instance from each of the following labels: • Label 0 – most common class; • Label 20 – second most common class; • Label 17 – least common class.



(a) Stability (S). (b) Fidelity (R^2). (c) Cosine Similarity (CS).

Figure 1: Average Explanation Stability (S), Fidelity (R^2) and Cosine Similarity (CS) across classification and LIME surrogate models.

The value of cosine similarity lies within the interval $[-1, 1]$. A score of 1 indicates that the surrogate model's outputs are perfectly aligned in direction with those of the black-box model, meaning the surrogate effectively captures the pattern of variation in the predictions. A value of 0 suggests no linear alignment (i.e., the prediction trends are orthogonal), while negative values denote an inverse relationship in the directional trends. In the realm of explainable AI, a high cosine similarity implies that the surrogate explanation preserves the underlying trend of the original model's behavior across the sampled instances. It is important to note that while cosine similarity highlights the consistency in the direction of variations, it is insensitive to differences in the absolute scale of the predictions. Therefore, it serves as a complementary measure to metrics like R^2, jointly providing a broader understanding of the surrogate's performance in approximating the black-box model.

4.4. Experiment methodology

Table 4 outlines the experimental variables. The experimental constants include employing Ridge regression, DT regressor and RF regressor as the surrogate models for LIME, generating 5000 perturbed samples for each explanation, focusing on the top 10 features for analysis, and running 30 explanation iterations for each configuration.

For each combination of test instance (3 labels), classification model (4 types), surrogate model architecture (3 types), explanations were generated and analyzed over 30 independent runs. Then each set of explanations was evaluated by calculating fidelity (R^2), Cosine Similarity (CS) and stability (S) metrics.

5. Results

In this section, we assess the effects of different surrogate regressors (DT, RF, and Ridge) and perturbation distributions (Beta, Gamma, Gaussian, Pareto, Weibull) on three core explainability metrics: fidelity (measured by R^2), stability (measured by the Jaccard similarity of the top 10 features across repeated runs), and directional alignment (measured by Cosine similarity). As reported in Figure 1b, fidelity varies considerably with both the surrogate architecture and the perturbation method. Notably, LIME with a Decision Tree surrogate achieves near-perfect fidelity ($R^2 \approx 1.000$) under Beta, Gamma, and Gaussian sampling across all four classifiers (DT, kNN, RF, XGBoost). Meanwhile, the Random Forest surrogate model also helps LIME attain high R^2 values (often above 0.90) but dips in certain Gaussian settings (e.g., $R^2 \approx 0.85$ for Decision Tree and XGBoost). By contrast, Ridge struggles with complex decision boundaries, sometimes dropping below 0.20 in R^2 (e.g., 0.047 for XGBoost under Gaussian sampling). Turning to stability in Figure 1a, Pareto and Beta commonly produce the most consistent top features, with some pairings (e.g., kNN with a Decision Tree surrogate under Pareto perturbations) reaching Jaccard similarities as high as 0.9901. Gaussian sampling proves more variable; for instance, a Decision Tree surrogate yields only 0.7095 stability for a Decision Tree classifier under Gaussian sampling, which is notably lower relative to other distributions. Additionally, Figure 1c summarize how well surrogate models capture the directional trends in black-box predictions via Cosine similarity. Here again, Pareto and Beta exhibit comparatively higher scores, indicating a better match between the surrogate's output pattern and the black-box classifier. The linear Ridge surrogate can still achieve strong Cosine similarity in certain cases (e.g., upwards of 0.95 for XGBoost with Beta), but it remains inconsistent for non-linear models, mirroring the drop seen in fidelity. Overall, these findings underscore that (1) aligning surrogate complexity with the underlying black-box model is crucial for strong local fidelity and robust directional alignment, and (2) selecting perturbation distributions closely approximating the data manifold (Pareto, Beta, Gamma) can substantially boost both consistency and directional fit of feature importance. Consequently, switching from the default Ridge-based LIME to a tree-based surrogate—especially when combined with Pareto or Beta perturbations—emerges as a promising strategy to generate more reliable and faithful local explanations in intrusion detection scenarios.

6. Conclusion

This work presents a systematic evaluation of how surrogate architecture (linear vs. tree-based) and perturbation distribution (Beta, Gamma, Gaussian, Pareto, Weibull) jointly impact LIME's ability to produce faithful, stable, and directionally aligned explanations for intrusion detection. Our findings confirm that `Decision Tree` and `Random Forest` surrogate models consistently outperform `Ridge Regression` in local fidelity, especially for complex classifiers like Random Forest and XGBoost, with R^2 values approaching or reaching 1.0 under distributions such as Beta, Gamma, and Pareto. At the same time, Pareto, Beta and Gamma also give significantly higher cosine similarity scores for many pairs of substitutes and classifiers, indicating greater consistency in the trend of forecasts compared to linear substitutes. These results confirm that tree-based surrogates better represent complex decision boundaries, while highlighting that the distribution of disturbances plays an important role in both accuracy and consistency. We also note that higher model complexity may reduce the direct interpretability of random forest surrogates, so the influence of different models needs to be rationally assessed and experimentally tested. Moreover, for further research we propose to expand the variety of labels from which the random test samples are picked to gain the generalized insights of explanation stability. Overall, our study shows that combining tree surrogates with carefully chosen sampling distributions can significantly increase the accuracy, stability and directional consistency of LIME explanations. Thus, we propose a more robust set of tools for cybersecurity practitioners to understand and justify model predictions.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] M. T. Ribeiro, S. Singh, C. Guestrin, "why should i trust you?": Explaining the predictions of any classifier, 2016. URL: https://arxiv.org/abs/1602.04938. arXiv:1602.04938.

[2] C. Molnar, 9.2 Local Surrogate (LIME) | Interpretable Machine Learning, URL: https://christophm.github.io/interpretable-ml-book/lime.html.

[3] G. Visani, E. Bagli, F. Chesani, A. Poluzzi, D. Capuzzo, Statistical stability indices for lime: Obtaining reliable explanations for machine learning models, Journal of the Operational Research Society 73 (2021) 91–101. URL: http://dx.doi.org/10.1080/01605682.2020.1865846. doi:10.1080/01605682.2020.1865846.

[4] X. Li, H. Xiong, X. Li, X. Zhang, J. Liu, H. Jiang, Z. Chen, D. Dou, G-lime: Statistical learning for local interpretations of deep neural networks using global priors, Artificial Intelligence 314 (2023) 103823. URL: https://www.sciencedirect.com/science/article/pii/S0004370222001631. doi:https://doi.org/10.1016/j.artint.2022.103823.

[5] T. A. A. Abdullah, M. S. M. Zahid, W. Ali, S. U. Hassan, B-lime: An improvement of lime for interpretable deep learning classification of cardiac arrhythmia from ecg signals, Processes 11 (2023). URL: https://www.mdpi.com/2227-9717/11/2/595. doi:10.3390/pr11020595.

[6] M. R. Zafar, N. M. Khan, Dlime: A deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems, 2019. URL: https://arxiv.org/abs/1906.10263. arXiv:1906.10263.

[7] Z. Tan, Y. Tian, J. Li, GLIME: General, Stable and Local LIME Explanation (????).

[8] D. Garreau, U. von Luxburg, Explaining the explainer: A first theoretical analysis of lime, 2020. URL: https://arxiv.org/abs/2001.03447. arXiv:2001.03447.

[9] J. Ribeiro, L. Cardoso, V. Santos, E. Carvalho, N. Carneiro, R. Alves, How reliable and stable are explanations of xai methods?, 2024. URL: https://arxiv.org/abs/2407.03108. arXiv:2407.03108.

[10] Z. Zhou, G. Hooker, F. Wang, S-lime: Stabilized-lime for model explanation, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, KDD '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 2429–2438. URL: https://doi.org/10.1145/3447548.3467274. doi:10.1145/3447548.3467274.

[11] X. Zhao, W. Huang, X. Huang, V. Robu, D. Flynn, Baylime: Bayesian local interpretable model-agnostic explanations, in: C. de Campos, M. H. Maathuis (Eds.), Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence, volume 161 of *Proceedings of Machine Learning Research*, PMLR, 2021, pp. 887–96. URL: https://proceedings.mlr.press/v161/zhao21a.html.

[12] G. Visani, E. Bagli, F. Chesani, Optilime: Optimized lime explanations for diagnostic computer algorithms, 2022. URL: https://arxiv.org/abs/2006.05714. arXiv:2006.05714.

[13] R. P. Bora, P. Terhörst, R. Veldhuis, R. Ramachandra, K. Raja, Slice: Stabilized lime for consistent explanations for image classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 10988–10996.

[14] S. M. Shankaranarayana, D. Runje, Alime: Autoencoder based approach for local interpretability, 2019. URL: https://arxiv.org/abs/1909.02437. arXiv:1909.02437.

[15] S. Shi, Y. Du, W. Fan, An extension of lime with improvement of interpretability and fidelity, 2020. URL: https://arxiv.org/abs/2004.12277. arXiv:2004.12277.

[16] M. Cinquini, R. Guidotti, Causality-aware local interpretable model-agnostic explanations, in: L. Longo, S. Lapuschkin, C. Seifert (Eds.), Explainable Artificial Intelligence, Springer Nature Switzerland, Cham, 2024, pp. 108–124.

[17] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani, A realistic cyber defense dataset (cse-cic-ids2018), 2018, 2018.

[18] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, in: International Conference on Information Systems Security and Privacy, 2018. URL: https://api.semanticscholar.org/CorpusID:4707749.

[19] J. L. Leevy, T. M. Khoshgoftaar, A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 Big Data, Journal of Big Data 7 (2020) 104. URL: https://doi.org/10.1186/s40537-020-00382-x. doi:10.1186/s40537-020-00382-x.

[20] L. D'hooge, T. Wauters, B. Volckaert, F. De Turck, Inter-dataset generalization strength of supervised machine learning methods for intrusion detection, Journal of Information Security and Applications 54 (2020) 102564. URL: https://www.sciencedirect.com/science/article/pii/S2214212619310415. doi:https://doi.org/10.1016/j.jisa.2020.102564.

[21] K. S. Huancayo Ramos, M. A. Sotelo Monge, J. Maestre Vidal, Benchmark-based reference model for evaluating botnet detection tools driven by traffic-flow analytics, Sensors 20 (2020). URL: https://www.mdpi.com/1424-8220/20/16/4501. doi:10.3390/s20164501.

[22] G. Karatas, O. Demir, O. K. Sahingoz, Increasing the performance of machine learning-based idss on an imbalanced and up-to-date dataset, IEEE Access 8 (2020) 32150–32162. doi:10.1109/ACCESS.2020.2973219.

[23] Q. R. S. Fitni, K. Ramli, Implementation of ensemble learning and feature selection for performance improvements in anomaly-based intrusion detection systems, in: 2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), 2020, pp. 118–124. doi:10.1109/IAICT50021.2020.9172014.

[24] Error Prevalence in NIDS datasets | CIC-IDS 2017, ???? URL: https://intrusion-detection.distrinet-research.be/CNS2022/CSECICIDS2018.html.

[25] M. Bacevicius, A. Paulauskaite-Taraseviciene, G. Zokaityte, L. Kersys, A. Moleikaityte, Comparative analysis of perturbation techniques in lime for intrusion detection enhancement, Machine Learning and Knowledge Extraction 7 (2025). URL: https://www.mdpi.com/2504-4990/7/1/21. doi:10.3390/make7010021.