R-LIME: Rectangular Constraints and Optimization for Local Interpretable Model-agnostic Explanation Methods

Genji Ohara[1][0009−0000−5854−2820], Keigo Kimura[1][0000−0002−3614−6568], and Mineichi Kudo[1][0000−0003−1013−3870]

Division of Computer Science and Information Technology
Graduate School of Information Sci. and Tech., Hokkaido University
Sapporo 060-0814, JAPAN,
`{genji-ohara, kimura5, mine}@ist.hokudai.ac.jp`

Abstract. In recent years, complex machine learning models such as deep neural networks have been used in various industrial fields due to their high accuracy. However, its complexity has been a major obstacle to implementation in decision-making situations where transparency of the decision process is required. In order to solve this problem, various post-hoc explanation methods have been proposed, but they have not been able to achieve interpretability of both the explanation and its scope. We propose a new method that interprets complex models in an interpretable approximation region. R-LIME locally approximizes a complex decision boundary with a linear classifier and maximizes the approximation region as long as the accuracy of the linear model is higher than a specified threshold. We demonstrate the effectiveness of the proposed method through a qualitative and quantitative evaluation using a real-world dataset. Finally, we discuss limitations and future work of the proposed method.

Keywords: Interpretable machine learning · Local surrogate model

1 Introduction

Machine learning models, such as deep learning and random forests, have been widely employed in various industrial applications due to their significant improvement in accuracy in recent years. However, the increasing complexity and black-box nature of these models pose challenges, particularly in critical decision-making scenarios like healthcare and finance, where the opacity of decision rationale becomes a major obstacle to implementation. Consequently, there has been extensive research in the field of post hoc explanation for machine learning models [5, 6, 4, 1].

Existing studies on post hoc explanations categorize into model-dependent and model-agnostic methods, depending on their dependence on the model's structure. Model-dependent methods primarily focus on deep neural networks and explain the model's behavior using internal parameters. While they allow a

direct approach to the model, applying the same method to models with different structures is often challenging. On the other hand, model-agnostic methods do not rely on internal parameters and utilize only the model's output. Although they provide high generality, their information is constrained, making them more versatile. Model-agnostic methods further classify into global and local methods based on their treatment of input space. Global methods aim to explain the model's behavior across the entire input space, seeking explanations valid for any input. However, providing global explanations becomes challenging as the model's complexity increases. Local methods, in contrast, explain the model's output for a specific input (focus point) in the vicinity of that input. While local methods offer simple and accurate explanations compared to global methods, their applicability is limited.

This paper specifically focuses on model-agnostic and local methods. For instance, Local Interpretable Model-agnostic Explanations (LIME) [5] approximates the complex model's decision boundary locally using a simpler model. Although LIME enables users to interpret the local behavior of complex models, it has been criticized for not optimizing the approximation region [4] and not presenting the approximation region in an interpretable form [6].

To address the limitations of LIME, we propose an algorithm that interprets complex models in an interpretable approximation region, taking inspiration from another explanation method called Anchor [6]. Our proposed method maximizes the accuracy of the learned approximation model within a rectangular approximation region under lower-bound constraints. The resulting rectangular region is expressed as a conjunction of predicates related to features, enhancing interpretability for the user.

2   Related Work

The figures presented visually compare three methods: LIME, Anchor, and BELLA.

In this chapter, we provide an overview of existing research on model-agnostic and local post hoc explanations, particularly focusing on studies closely related to the proposed method.

### 2.1   LIME (Local Interpretable Model-agnostic Explanations) [5]

LIME is a method for locally approximating a trained black-box classifier $f : \mathbb{R}^d \to \{0, 1\}$ around a focal point $x \in \mathbb{R}^d$ using a linear model $g : \mathbb{R}^d \to \{0, 1\}$ (fig. 2(a)). The process involves:

1. Obtaining a set of perturbed samples $\mathcal{Z}_p$ around x and the set of pseudo-labels $f(\mathcal{Z}_p) = \{f(z) \mid z \in \mathcal{Z}_p\}$.
2. Learning an interpretable model g using $\mathcal{Z}_p$ and $f(\mathcal{Z}_p)$ by minimizing the following loss function:

$$\mathcal{L}(f, g, \pi_x) = \sum_{z \in \mathcal{Z}_p} \pi_x(z)(f(z) - g(z))^2, \tag{1}$$

(a) Instance around the focal point.
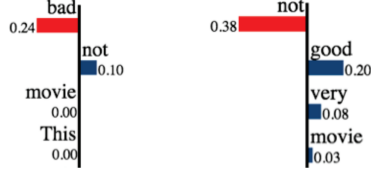
(b) Explanation by LIME.

(c) Explanation by Anchor.

Fig. 1: Examples of output results by LIME and Anchor [6].

where $\pi_x(z)$ is a weight function designed to be larger for samples closer to x, typically implemented using an exponential kernel.

LIME provides valuable insights into the local behavior of the model by showing the contribution of each feature to the output $f(x)$. However, due to not explicitly indicating the perturbation region, users cannot assess the effective scope of the explanation [6]. An example explanation for an LSTM sentiment prediction model using LIME is illustrated in fig. 1(b). The left explanation suggests that the word "not" contributes to the model's positive prediction, which does not apply to the instance on the right. However, without the explicit perturbation region, users might mistakenly apply the left explanation to the right instance, potentially leading to a misunderstanding of the black-box model's behavior [6].

2.2 Anchor [6]

Anchor represents a rectangular region, expressed as a conjunction of predicates (rules) related to features, that maximizes the probability of the black-box classifier f outputting $f(x)$ within the region. It aims to highlight important features contributing significantly to the output (fig. 2(b)).

For a discrete m-dimensional input space $\mathbb{D}^m$ with a trained black-box classifier $f : \mathbb{D}^m \to \{0, 1\}$, an instance $x \in \mathbb{D}^m$, and a distribution $\mathcal{D}$ over the input space, a rule $A(z) = a_{i_1}(z) \wedge a_{i_2}(z) \wedge \cdots \wedge a_{i_t}(z)$ is defined. The predicates $a_i(z)$ evaluate to true $(= 1)$ when $z_i = x_i$ and false $(= 0)$ otherwise. The precision acc(A) and coverage cov(A) of the rule A are defined as follows:

$$\text{acc}(A) = \mathbb{E}_{z \sim \mathcal{D}(z|A)}[\mathbb{1}_{f(z)=f(x)}], \tag{2}$$

$$\text{cov}(A) = \mathbb{E}_{z \sim \mathcal{D}(z)}[A(z)]. \tag{3}$$

Here, $\mathcal{D}(z|A)$ is the conditional distribution in the region where the rule A is true. $\mathrm{acc}(A)$ represents the probability that the output of f matches between the perturbation z and the focal point x in the region A, and $\mathrm{cov}(A)$ expresses the probability that the perturbation z fits into A.

An Anchor maximizes coverage within the given threshold τ while maintaining precision. However, direct computation of eq. (2) is not feasible, and it is evaluated using PAC (Probably Approximately Correct) through sampling. Introducing a confidence level $1 - \delta$ ($0 \leq \delta \leq 1$), the precision constraint is relaxed as follows:

$$P(\mathrm{acc}(A) \geq \tau) \geq 1 - \delta. \tag{4}$$

Thus, the following optimization problem is solved:

$$A^* = \underset{A \text{ s.t. } P(\mathrm{acc}(A) \geq \tau) \geq 1 - \delta \wedge A(x) = 1}{\arg\max} \mathrm{cov}(A). \tag{5}$$

While Anchor provides a clear presentation of the effective scope of the explanation, the information it includes is less than LIME, limiting its utility [6]. An example explanation for an LSTM sentiment prediction model using Anchor is depicted in fig. 1(c). The left explanation suggests that changing words other than "not" and "bad" has little impact on the classifier's output. While it clearly does not apply to the right instance, the explanation does not provide details about the influence of words like "not" or "bad," resulting in less user insight into the model's behavior compared to LIME (fig. 1(b)).

2.3 BELLA (Black box model Explanations by Local Linear Approximations) [4]

BELLA addresses the non-optimization of the approximation region, a drawback of LIME, by learning a local linear approximation model g using a subset Z' of the dataset Z instead of perturbed samples (fig. 2(c)). The process involves:

1. Computing the distance $d_x(z)$ for all instances z in Z with respect to the focal point x.
2. Selecting the k instances with the smallest $d_x(z)$, forming a subset $Z_s(k) \subseteq Z$.
3. Learning the local linear approximation model g using $Z_s(k)$ and calculating the similarity measure $\mathcal{R}(k)$ between f and g.
4. Obtaining the optimal k^* by maximizing $\mathcal{R}(k)$.

BELLA aims to enhance the reliability of explanations by optimizing the approximation region for high-accuracy linear approximations. However, it assumes direct access to the dataset Z used for training the classifier, making it inapplicable in scenarios where direct access to the dataset is restricted for privacy reasons. Additionally, as BELLA presents the approximation region as a subset of the dataset, its interpretability is lower compared to Anchor.

(a) LIME: Locally approximates decision boundaries around the focal point.

(b) Anchor: Maximizes coverage of a rectangular region containing the focal point under precision constraints.

(c) BELLA: Explores a subset of the dataset, maximizing similarity between the local linear approximation model and the black-box classifier.

(d) R-LIME: Maximizes coverage of a rectangular region containing the focal point under lower constraints on the accuracy of the approximation model.
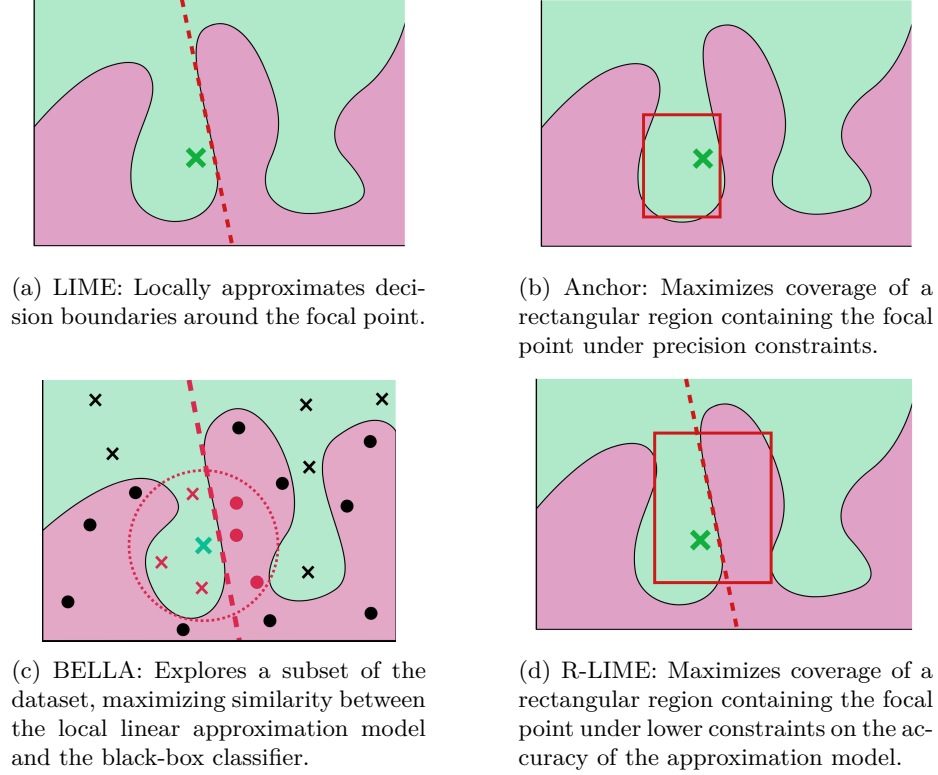
Fig. 2: Visual comparison of LIME, Anchor, BELLA and R-LIME (our proposed method). The solid line represents the rectangular region containing the focal point, and the dashed line represents the learned approximation model.

3   Proposed Method

3.1   Overview

The proposed method aims to overcome the limitations of LIME [5] and its improvement methods such as Anchor [6] and BELLA [4]. Similar to LIME, the proposed method locally approximates the given black-box classifier f around the focal point x using a linear model g. However, by adopting a rectangular perturbation region similar to Anchor, it represents predicates (rules) concerning features. The accuracy of explanations is defined as the "accuracy" of rules, and the "coverage" of rules defines the generality of explanations. The proposed method maximizes coverage under lower constraints on accuracy (fig. 2(d)).

Anchor maximizes the coverage of region A within which the output of the black-box classifier f matches $f(x)$ with respect to the probability under lower constraints. The proposed method, on the other hand, learns an approximation model g within the rectangular region A and maximizes the coverage of A under
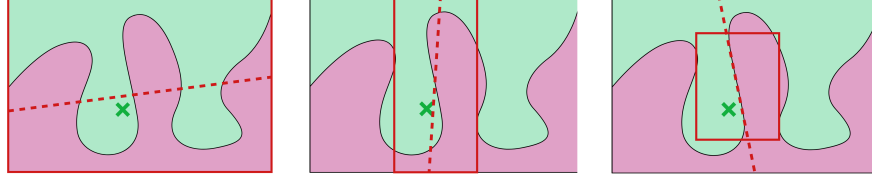
Fig. 3: Overview of the R-LIME algorithm. The progression of the algorithm from left to right is illustrated. The solid red line represents the rectangular region A, and the red dashed line represents the linear approximation model g learned within A. The initial value of A is an empty rule (entire input space), and predicates are added to A, reducing coverage. The process continues until $\mathrm{acc}(A) \geq \tau$, at which point the rule with the maximum coverage is output.

lower constraints on the accuracy of g. We modify the definition of accuracy using eq. (2) as follows:

$$\mathrm{acc}(A) = \max_{g \in G} \mathbb{E}_{z \sim \mathcal{D}(z|A)}[\mathbb{1}_{f(z)=g(z)}]. \tag{6}$$

Here, G represents the set of possible linear approximation models. By solving the optimization problem in eq. (5) under this modified accuracy definition, we can select rules that learn high-accuracy approximation models, among which the coverage is maximized. The proposed method's algorithm, shown in section 3.2, is largely based on the algorithms used in Anchor.

In this paper, we refer to the proposed method as "R-LIME" (Ruled LIME). The characteristics of R-LIME are listed below:

- The perturbation vector's generation region is optimized and presented in a user-interpretable format.
- Interpretation is conducted without direct access to the dataset, utilizing only the distribution.
- The number of samples is dynamically determined based on the estimated accuracy without the need for a predefined amount.

3.2   Algorithm

For non-convex optimization problems like eq. (5), greedy methods are often employed. However, greedy methods tend to converge to local optima, and to address this, the proposed method utilizes beam search, which selects multiple candidates at each iteration. The pseudocode is shown in algorithm 1.

Candidate Rule Generation To generate new candidate rules, one additional predicate is added to each of the B candidate rules obtained in the previous iteration. The pseudocode is shown in algorithm 2. Here, $T(x)$ represents the set of attribute-value pairs (predicates) that x possesses, and $T(x) \setminus A$ represents the conjunction of predicates in $T(x)$ not included in rule A.

Algorithm 1 R-LIME

Input: Black-box model f, Target instance x, Distribution $\mathcal{D}$, Threshold τ, Beam width B, Tolerance ϵ, Confidence level $1 - \delta$
Output: Rule A^* satisfying Eq. (5)
1:   $A^* \leftarrow$ **null**, $\mathcal{A}_0 \leftarrow \emptyset$, $t \leftarrow 0$ ▷ Initialize the set of candidate rules $\mathcal{A}_0$ to $\emptyset$
2:   **while** $A^* =$ **null do**
3:      $t \leftarrow t + 1$
4:      $\bar{\mathcal{A}}_t \leftarrow \text{GENERATECANDS}(\mathcal{A}_{t-1})$ ▷ Generate new candidate rules from $\mathcal{A}_{t-1}$
5:      $\mathcal{A}_t \leftarrow \text{B-BESTCANDS}(\bar{\mathcal{A}}_t, \mathcal{D}, B, \epsilon, \delta)$ ▷ Select B best candidate rules from $\bar{\mathcal{A}}_t$
6:      $A^* \leftarrow \text{LARGESTCAND}(\mathcal{A}_t, \tau, \delta)$ ▷ Select the largest rule satisfying Eq. (5)
7:   **end while**

Algorithm 2 Generating candidate rules

1:   **function** $\text{GENERATECANDS}(\mathcal{A}, x)$
2:      **if** $\mathcal{A} = \emptyset$ **then return** $\{\emptyset\}$
3:      $\bar{\mathcal{A}} \leftarrow \emptyset$
4:      **for all** $A \in \mathcal{A}$ **do**
5:         **for all** $a_i \in (T(x) \setminus A)$ **do**
6:            $\bar{\mathcal{A}} \leftarrow \bar{\mathcal{A}} \cup (A \wedge a_i)$
7:         **end for**
8:      **end for**
9:      **return** $\bar{\mathcal{A}}$
10: **end function**

Algorithm 3 Selecting B best rules with highest accuracy (KL-LUCB [2])

1:   **function** $\text{B-BESTCANDS}(\bar{\mathcal{A}}, \mathcal{D}, B, \epsilon, \delta)$
2:      **initialize** $\text{acc}, \text{acc}_u, \text{acc}_l$ for $\forall A \in \bar{\mathcal{A}}$
3:      $\mathcal{A} \leftarrow \text{B-PROVISIONALLYBESTCANDS}(\bar{\mathcal{A}})$ ▷ B rules with highest accuracy
4:      $A \leftarrow \arg\min_{A \in \mathcal{A}} \text{acc}_l(A, \delta)$ ▷ The rule with the smallest lower bound
5:      $A' \leftarrow \arg\max_{A' \notin (\bar{\mathcal{A}} \setminus \mathcal{A})} \text{acc}_u(A', \delta)$ ▷ The rule with the largest upper bound
6:      **while** $\text{acc}_u(A', \delta) - \text{acc}_l(A, \delta) > \epsilon$ **do**
7:         **sample** $z \sim \mathcal{D}(z|A), z' \sim \mathcal{D}(z'|A')$
8:         **update** $\text{acc}, \text{acc}_u, \text{acc}_l$ for A and A'
9:         $\mathcal{A} \leftarrow \text{B-PROVISIONALLYBESTCANDS}(\bar{\mathcal{A}})$
10:       $A \leftarrow \arg\min_{A \in \mathcal{A}} \text{acc}_l(A, \delta)$
11:       $A' \leftarrow \arg\max_{A' \notin (\bar{\mathcal{A}} \setminus \mathcal{A})} \text{acc}_u(A', \delta)$
12:      **end while**
13:      **return** $\mathcal{A}$
14: **end function**

Algorithm 4 Selecting the largest rule satisfying the constraint

---

1:   **function** $\text{LARGESTCAND}(\mathcal{A}, \tau, \delta)$
2:      $A^* \leftarrow$ **null**
3:      **for all** $A \in \mathcal{A}$ s.t. $\text{acc}_l(A, \delta) > \tau$ **do**
4:         **if** $\text{cov}(A) > \text{cov}(A^*)$ **then** $A^* \leftarrow A$
5:      **end for**
6:      **return** A^*
7:   **end function**

Selection of Candidate Rule with Maximum Accuracy Given the set of generated candidate rules $\bar{\mathcal{A}}$, the algorithm selects the top B candidates with the highest accuracy. This is treated as an optimal arm identification problem in the multi-armed bandit framework. Each candidate rule $A_i \in \bar{\mathcal{A}}$ is considered an arm, and their accuracy $\mathrm{acc}(A_i)$ is treated as the distribution of rewards. By sampling $z \sim \mathcal{D}(\cdot|A_i)$ and obtaining the reward $\mathbb{1}_{f(z)=g_i(z)}$ for each trial, the algorithm updates g_i using the sampled perturbation vector z and the pseudo-label $f(z)$ immediately after each trial. To efficiently select the rule (arm) with the highest accuracy, the KL-LUCB algorithm [2] is employed. The pseudocode is shown in algorithm 3. The KL-LUCB algorithm assumes that the reward distribution for each arm remains unchanged, while the proposed method updates the approximation model g_i with each sampling, which may not satisfy this assumption. This issue is discussed further in section 5.2.

Selection of Rule with Maximum Coverage Meeting Constraints To meet the constraint in eq. (4), rule A must satisfy the lower constraint

$$\mathrm{acc}_l(A, \delta) > \tau \tag{7}$$

where $\mathrm{acc}_l(A, \delta)$ is the lower bound on accuracy and δ is a small positive value. To check this condition, the algorithm performs a binary search on δ to find the smallest δ that satisfies eq. (7). The pseudocode is shown in algorithm 4.

The algorithmic procedure outlined above approximates the solution to the optimization problem in eq. (5) under the definition of accuracy in eq. (6). It iteratively selects a rectangular region (rule) with the learned approximation model's accuracy exceeding a given threshold while maximizing coverage. The selected rule, with the maximum coverage meeting accuracy constraints, is output after each iteration.

4   Experiments

To verify the effectiveness of the proposed method, LIME and R-LIME were compared using a single dataset.

4.1   Qualitative Evaluation

Experimental Setup The experiments utilized the recidivism dataset [7]. This dataset contains information on 9,549 prisoners released from North Carolina prisons between July 1, 1979, and June 30, 1980. The dataset includes 19 items such as race, gender, presence of alcohol dependence, number of prior offenses, and recidivism. For this experiment, we treated the binary classification problem of predicting the presence or absence of recidivism (Recidivism) as the target label. Preprocessing included discretization of continuous features, resulting in 15 features. This problem setting simulates a case where a machine learning

Table 1: Attributes and overview of the recidivism dataset used in the experiments. Continuous features are discretized, and only binary and ordinal features are considered.

Attribute	Overview	# of Possible Values
Race	Race (Black or White)	2
Alcohol	Presence of serious alcohol issues	2
Junky	Drug usage	2
Supervised Release	Supervised release	2
Married	Marital status	2
Felony	Felony or not	2
WorkRelease	Participation in work release program	2
Crime against Property	Crime against property or not	2
Crime against Person	Crime against a person or not	2
Gender	Gender (Female or Male)	2
Priors	Number of prior offenses	4
YearsSchool	Years of formal education completed	4
PrisonViolations	Number of prison rule violations	3
Age	Age	4
MonthsServed	Months served in prison	4
Recidivism	Recidivism or not	2

model is introduced to decide parole for prisoners. Since such decisions can have a significant impact on a person's life, it is crucial for users to interpret the outputs of black-box models appropriately.

The dataset, after removing missing values, was split into training data (7,639 instances) and test data (955 instances). A random forest model with 50 trees was trained using the training data. Subsequently, LIME and R-LIME explanations were generated for two instances extracted from the test data (fig. 4). In R-LIME, logistic regression was used as the linear approximation model, and the distribution $\mathcal{D}$ was estimated using a multivariate normal distribution based on the training data.

The beam width was set to $B = 10$, the confidence coefficient to $1 - \delta = 0.95$, and the tolerance of the KL-LUCB algorithm to $\epsilon = 0.05$. Accuracy thresholds τ were set to $\tau = 0.70, 0.80, 0.90$.

Experimental Results The results of the experiment are shown in figs. 5 and 6. The values assigned to each attribute name represent the contribution (weights of the learned linear approximation model) to the output of the black-box classifier, normalized such that the absolute sum is 1. The figures also highlight the five attributes with the highest absolute contribution.

Explanations generated by LIME (figs. 5(a) and 6(a)) indicate that attributes such as having a significant prior record (Prior) or committing a crime against property (Crime against Property) primarily contribute positively to the pre-

Race	Black (0)
Alcohol	No (0)
Junky	No (0)
Supervised Release	Yes (1)
Married	Yes (1)
Felony	No (0)
WorkRelease	Yes (1)
Crime against Property	No (0)
Crime against Person	No (0)
Gender	Male (1)
Priors	1
YearsSchool	8.00 < YearsSchool <= 10.00 (1)
PrisonViolations	0
Age	Age > 33.00 (3)
MonthsServed	4.00 < MonthsServed <= 9.00 (1)
Recidivism	No more crimes (0)

(a) Instance A

Race	Black (0)
Alcohol	Yes (1)
Junky	No (0)
Supervised Release	Yes (1)
Married	No (0)
Felony	No (0)
WorkRelease	Yes (1)
Crime against Property	Yes (1)
Crime against Person	No (0)
Gender	Male (1)
Priors	1
YearsSchool	YearsSchool > 11.00 (3)
PrisonViolations	0
Age	21.00 < Age <= 26.00 (1)
MonthsServed	4.00 < MonthsServed <= 9.00 (1)
Recidivism	Re-arrested (1)

(b) Instance B

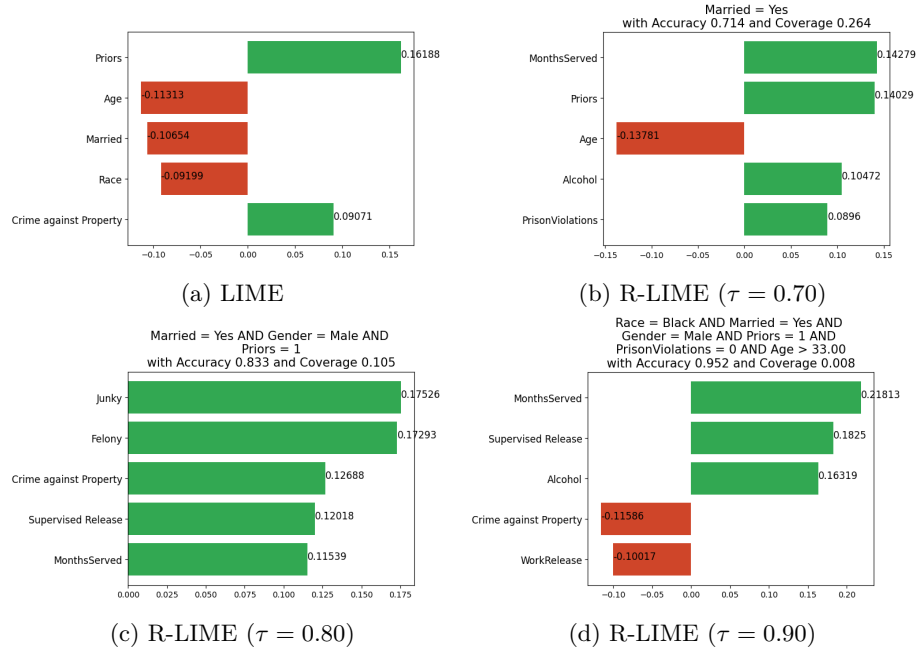Fig. 4: Two instances sampled from recidivism dataset.

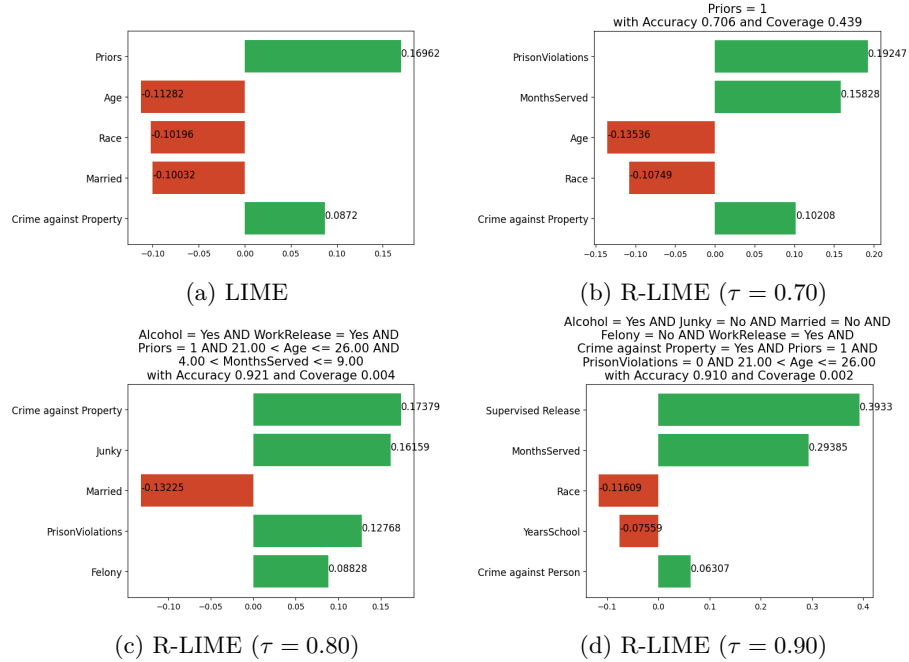Fig. 5: Explanation for Instance A by LIME and R-LIME.

Fig. 6: Explanation for Instance B by LIME and R-LIME

diction (predicting that the inmate will be re-incarcerated). On the other hand, attributes like older age (Age), being married (Married), and being of white race (Race) contribute negatively to the prediction (predicting that the inmate will not be re-incarcerated). While these LIME explanations provide valuable insights into the behavior of the black-box model, they do not explicitly indicate the range of application, leaving users unable to determine to which inmates the explanations are applicable.

In contrast, R-LIME expresses the application range of explanations as a conjunction of predicates. For example, the explanation for instance A under $\tau = 0.70$ (fig. 5(b)) indicates that it is applicable only to married males (Married=Yes, Gender=Male). Furthermore, the generated explanations include the accuracy and coverage of the approximation region, allowing users to evaluate how reliable the explanations are. For example, the coverage of the explanation for instance B under $\tau = 0.90$ (fig. 6(d)) is 0.002, indicating that the decision boundaries around instance B are complex, making it challenging to obtain a high-precision linear approximation. This allows users to discern that the application scope of this explanation is very narrow, limiting its utility.

4.2   Quantitative Evaluation

Experimental Setup To demonstrate that R-LIME adapts a highly accurate linear approximation model to the optimized approximation region, we conducted a comparison of the local accuracy of explanations between LIME and R-LIME. Using the same settings as in section 4.1, we randomly sampled 100 instances from the test data of the recidivism dataset and generated explanations (with $\tau = 0.70, 0.80, 0.90$) using LIME and R-LIME. We then sampled 10,000 instances within the rectangular region obtained by R-LIME and calculated the local accuracy of both methods.

Experimental Results The results are presented in fig. 7, showcasing the distribution of the accuracy of the linear approximation models obtained by LIME and R-LIME. R-LIME exhibits higher accuracy compared to LIME for all values of τ. This suggests that the linear approximation models learned by LIME and R-LIME differ significantly, and R-LIME learns a high-precision linear approximation model adapted to the rectangular region. Additionally, as τ increases, the variability in the accuracy of LIME widens. This indicates that the linear models learned by LIME may not function effectively as approximation models depending on how the region is selected.

5   Challenges and Future Perspectives

5.1   Behavior Regarding Imbalanced Label Distribution

R-LIME may generate less useful explanations when there is bias in the distribution of black-box model outputs. When the distribution of black-box model
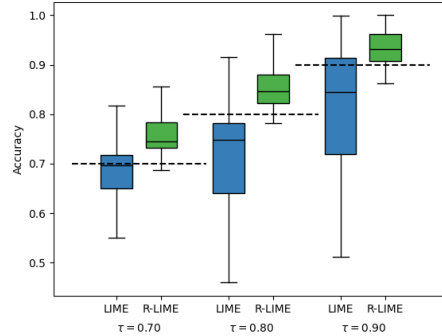
Fig. 7: Comparison of local accuracy between LIME and R-LIME. We generated explanations for 100 instances using each method and calculated the accuracy for each method within the 10,000 instances sampled inside the rectangular region obtained by R-LIME.

Table 2: Deviation between the estimated accuracy and the true accuracy.

	Estimation	True	Estimation−True
Average	.811	.829	.012
Standard Deviation	.018	.023	.017

outputs is significantly biased for a given accuracy threshold τ (the ratio of the majority label is less than τ), the approximation region generated by R-LIME covers the entire input space, and the learned linear approximation model always outputs the majority label (fig. 8(b)).

As a first solution to this problem, changing the loss function is considered. Using weighted logistic loss or Focal Loss [3] as the loss function might lead to the generation of more useful explanations in the case of imbalanced label distribution. Another solution involves adding constraints to limit the label distribution bias within the approximation region. In addition to eq. (4), adding a constraint like

$$\left(\mathbb{E}_{z \sim \mathcal{D}(z|A)}[\mathbb{1}_{f(z)=1}] - \frac{1}{2}\right)^2 < \mu \tag{8}$$

could suppress the excessive expansion of the approximation region.

5.2   Changes in Reward Distribution in Optimal Arm Identification

In R-LIME, the problem of selecting the rule with the highest accuracy is formulated as the optimal arm identification problem in multi-armed bandit theory, solved using the KL-LUCB algorithm [2]. However, this algorithm assumes that

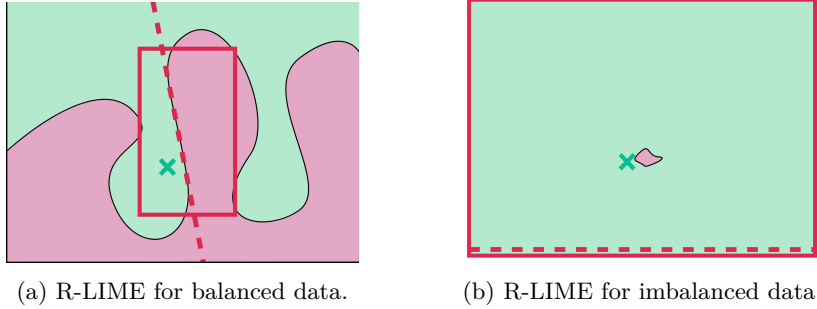(a) R-LIME for balanced data.   (b) R-LIME for imbalanced data.

Fig. 8: Behavior of R-LIME for balanced and imbalanced data. In case of imbalanced label distribution, the approximation region covers the entire input space and the linear approximation model always outputs the majority label.

the reward distribution remains constant, while in R-LIME, the reward distribution (accuracy of the linear approximation model) changes with every update of the approximation model after sampling. Therefore, rewards obtained at an early stage when the accuracy is still low might influence the estimated value and deviate from the true value.

An experiment was conducted to evaluate the deviation between the estimated accuracy and the true accuracy. Explanations were generated for 3,200 data instances sampled from the dataset, and the estimated accuracy was compared with the true accuracy. The true accuracy was calculated based on 1,000 instances sampled within the approximation region, considering cases where the outputs of the black-box model and the linear approximation model matched. The results in table 2 show a mean deviation of 0.012 with a standard deviation of 0.017. While the deviation is relatively small, improvements to the selection algorithm need to consider the changing accuracy.

5.3   Evaluation through User Studies

Given the nature of the field of "interpretable machine learning," where the goal is to "interpret" machine learning models, it is challenging to quantitatively compare various methods. While quantitative experiments were conducted in section 4.2, they mainly demonstrated that the proposed method generates locally accurate explanations compared to LIME. The results do not directly prove the high interpretability of the proposed method.

One potential method to quantitatively evaluate "interpretability" is through user studies. For example, [6] conducted user studies, assuming that if users could predict the output of a black-box model with high accuracy based on provided explanations, those explanations contained valuable information about the model's behavior. Similarly, conducting user studies in this research would be desirable to evaluate the interpretability of the proposed method.

6 Conclusion

We identified challenges in existing methods for local model-agnostic post-hoc explanations of black-box classifiers and proposed R-LIME to address them. We represented the rectangular region for local linear approximation as a conjunction of predicates of features and proposed an algorithm to maximize coverage under the constraint of minimum approximation accuracy. Comparing the outputs of LIME and R-LIME on real-world datasets, we demonstrated that explanations provided by R-LIME have clearer application scopes and can be evaluated by users for reliability and generality. However, we discussed the instability of behavior concerning imbalanced label distributions and raised questions about the theoretical validity of using the KL-LUCB algorithm.

References

1. Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., Giannotti, F.: Local rule-based explanations of black box decision systems (2018)
2. Kaufmann, E., Kalyanakrishnan, S.: Information complexity in bandit subset selection. In: Shalev-Shwartz, S., Steinwart, I. (eds.) Proceedings of the 26th Annual Conference on Learning Theory. Proceedings of Machine Learning Research, vol. 30, pp. 228–251. PMLR, Princeton, NJ, USA (12–14 Jun 2013), https://proceedings.mlr.press/v30/Kaufmann13.html
3. Lin, T.Y., Goyal, P., Girshick, R., He, K., DollÃąr, P.: Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(2), 318–327 (2020). https://doi.org/10.1109/TPAMI.2018.2858826
4. Radulovic, N., Bifet, A., Suchanek, F.: Bella: Black box model explanations by local linear approximations (2023)
5. Ribeiro, M.T., Singh, S., Guestrin, C.: "why should i trust you?": Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1135–1144. KDD '16, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939778, https://doi.org/10.1145/2939672.2939778
6. Ribeiro, M.T., Singh, S., Guestrin, C.: Anchors: High-precision model-agnostic explanations. Proceedings of the AAAI Conference on Artificial Intelligence **32**(1), 1527–1535 (Apr 2018). https://doi.org/10.1609/aaai.v32i1.11491, https://ojs.aaai.org/index.php/AAAI/article/view/11491
7. Schmidt, P., Witte, A.D.: Predicting Recidivism in North Carolina, 1978 and 1980. Inter-university Consortium for Political and Social Research (1988)