Reviewer Response

We would like to appriciate all the reviewers for their very helpful comments. We added an overview of other existing methods, discussion about parameter selection, and comparisons between Anchor and R-LIME. Below are suggestions from reviewers and our response.

Reviewer #1

Suggestions for Rebuttal:

1. Offer guidance on how to select optimal parameters for different datasets and models.

   Thank you for your suggestion. We added guidance on how parameters should be selected in practical use cases to section 5.

2. Include a discussion of other model-agnostic interpretability methods, such as SHAP, to provide a more comprehensive overview.

   Thank you for your suggestion. We added an overview of other model-agnostic interpretability methods to section 2.

3. Add experimental comparisons with more other model-agnostic interpretability methods to better evaluate R-LIME's effectiveness.

   Thank you for your suggestion. We additionally conducted qualitative and quantitative comparisons between Anchor and R-LIME, which demonstrated that R-LIME is more effective than Anchor.

Reviewer #4

The authors should kindly explain why there is not a quantitative comparison between their proposed method R-LIME and Anchor. The lack of the comparison makes it hard to evaluate the impact of the proposed model.

Thank you for your insightful comments.

Reviewer #5

– It would be beneficial to move the comparison of R-LIME with Anchor and LIME to the Related Work section for improved clarity and organization.

  Thank you for your suggestion.
  ...

– The authors should consider exploring a more compatible method than the KL-LUCB algorithm to achieve higher accuracy. Although the experimental

results suggest a minor effect, there may be potential underlying issues that need to be addressed.

Thank you for your valuable feedback. It is true that there are theoretical issues for using the KL-LUCB algorithm, although our results suggest that its practical effect is negligible. We leave the issues to the future work because of our limited time.

– The title of the 'Challenges and Future Work' section could be more aptly replaced with 'Discussion' to better reflect the content.

Thank you for your suggestion. We renamed the section to 'Discussion'.

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

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

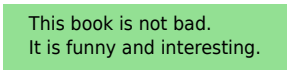
Division of Computer Science and Information Technology
Graduate School of Information Sci. and Tech., Hokkaido University
Sapporo 060–0814, JAPAN,
{genji-ohara, kimura5, mine}@ist.hokudai.ac.jp

Abstract. In recent years, complex machine learning models have been introduced in various industrial fields due to their high accuracy. However, their increasing complexity has been a major obstacle to implementation in sensitive decision-making situations. In order to address 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, R-LIME, which interprets a complex classifier in an interpretable scope. R-LIME locally and linearly approximates a complex decision boundary of a black-box classifier in a rectangular region and maximizes the region as long as the approximation accuracy exceeds a given threshold. The resulting rectangular region is interpretable for users because it is expressed as a conjunction of feature predicates. Through qualitative and quantitative comparisons with the existing method on a real-world dataset, we demonstrate that R-LIME provides more reliable and interpretable explanations than existing methods.

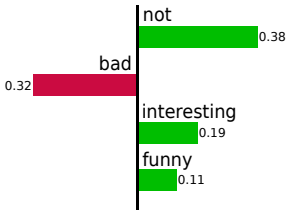
Keywords: Interpretable machine learning · Local surrogate model

1 Introduction

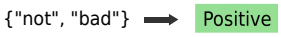
In recent years, complex machine learning models, such as deep neural networks and random forests, have been widely introduced in various industrial fields due to their significant improvement in accuracy. However, their increasing complexity and black-box nature pose challenges, particularly in critical decision-making scenarios such as healthcare and finance, where the opacity of decision process becomes a major obstacle to implementation. In order to address this problem, there has been extensive research in the field of post-hoc explanations for machine learning models [4, 9–11]. Existing post-hoc explanation methods are categorized into *model-dependent* and *model-agnostic* methods based on their dependence on the model's structure, and the latter are further classified into *global* and *local* methods based on their locality in input space [13].



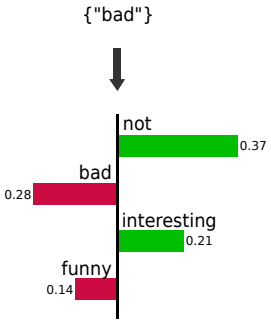
(a) Focal point.



(b) Explanation by LIME. It provides contributions of each feature to the output, but does not explicitly indicate its scope.



(c) Explanation by Anchor. It provides the effective scope of the explanation, but does not show the influence of each feature.



(d) Explanation by R-LIME. It provides both contributions of each feature and its effective scope.

Fig. 1: Example of explanations by LIME [10], Anchor [11] and R-LIME (our proposed method) for a sentiment prediction model.

In this paper, we focus on *local* and *model-agnostic* methods. LIME [10] and Anchor [11] are representative local model-agnostic methods. An example of explanations by LIME and Anchor for a sentiment prediction model is illustrated in Fig. 1. LIME linearly approximates the complex decision boundary around the given focal point (Fig. 1(a)), then provides the weights of the linear model as the contribution of each feature to the output. The explanation by LIME (Fig. 1(b)) suggests that the word "not" mainly contributes to the positive prediction, but does not explicitly indicate its effective scope. Without the scope, users might mistakenly apply the knowledge derived from the explanation to other instances far from the focal point, potentially leading to misunderstanding of the black-box model's behavior [11]. For this example, users may apply the derived insights to the sentence "This book is not good." and mistakenly conclude that the word "not" mainly contributes to the positive prediction for this sentence as well, which is obviously incorrect. Anchor maximizes the coverage of a rectangular region containing the focal point as long as the probability of the black-box classifier outputting the same label as the focal point within the region exceeds a given threshold. While Anchor provides an effective scope of the explanation, users can get less insight compared to LIME. The explanation by Anchor (Fig. 1(c)) suggests that replacing words other than "not" and "bad" has little impact on the classifier's output. While it clearly cannot be applied to the sentence "This book is not good" because of not including the word "bad", the explanation does not provide details about the influence of each word, resulting in less user insight into the model's behavior compared to LIME.

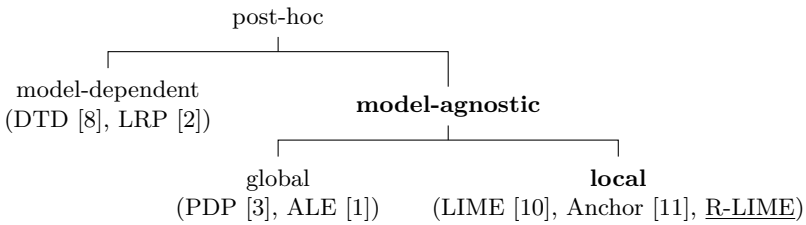


Fig. 2: Categorization of post-hoc explanation methods. We focus on *model-agnostic* and *local* methods, which explain model's local behavior using only its output.

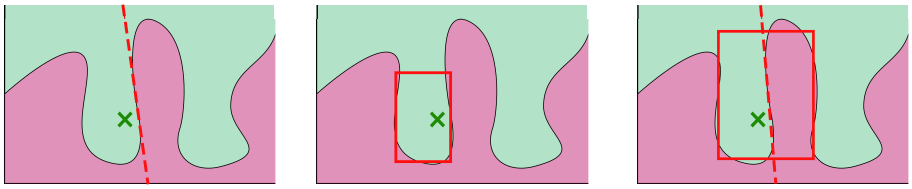
To address these limitations, we propose a new method called R-LIME (Ruled LIME), which provides both the contributions of each feature to the output and the effective scope of the explanation. R-LIME linearly approximates a complex decision boundary in a rectangular region and maximizes the region as long as the accuracy of the linear classifier exceeds a given threshold. The region is interpretable for users because it is expressed as a conjunction of feature predicates. An example of the explanation by R-LIME for a sentiment prediction model is shown in Fig. 1(d). It is clear that users can apply the insights derived from the explanation only to the sentences containing the word "not".

2 Related Work

In this section, we overview existing research on post-hoc explanation methods, which explain the behavior of black-box models already trained. As shown in Fig. 2, post-hoc methods are classified into several categories.

They are broadly divided into *model-dependent* and *model-agnostic* methods based on their dependence on the model's structure. Model-dependent methods, such as deep Taylor decomposition (DTD) [8] and layer-wise relevance propagation (LRP) [2], most of which focus on neural networks and explain the model's behavior using its parameters [13]. While these methods provide detailed explanations (e.g., layer-wise explanations for neural networks), it is often challenging to apply the same method to models with different structures. On the other hand, model-agnostic methods use only the output of the model. Although they are applicable to any model, they cannot explain the reasoning process inside the model.

Furthermore, model-agnostic methods are categorized into *global* and *local* methods based on their locality in input space. Global methods, such as partial dependence plots (PDP) [3] and accumulated local effects (ALE) [1], aim to explain the model's behavior across the entire input space. However, providing global explanations becomes challenging as the model's complexity increases. In contrast, local methods, such as local interpretable model-agnostic explanations (LIME) [10], Anchor [11] and shapley additive explanations (SHAP) [7], explain model's behavior in the vicinity of a specific input. While they offer explanations



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

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

(c) R-LIME: Maximizes coverage of a rectangular region under lower constraints on the accuracy of the linear classifier.

Fig. 3: Visual comparison of LIME, Anchor and R-LIME (our method). The dashed line represents the local linear approximation model, and the solid line represents the rectangular region containing the focal point.

more simple and accurate than global methods, the scope of the explanation is limited locally.

3 Proposed Method

3.1 Previous Work

We specifically focus on *local* and *model-agnostic* methods. In this section, we briefly review existing research on local model-agnostic explanations, particularly focusing on studies closely related to our proposed method.

LIME (Local Interpretable Model-agnostic Explanations) [10] LIME locally approximates a black-box classifier $f : \mathbb{R}^m \to \{0, 1\}$ around a focal point $x \in \mathbb{R}^m$ by a linear classifier $g : \mathbb{R}^m \to \{0, 1\}$ (Fig. 3(a)). The approximation is performed by the following steps:

1. Generating a set of perturbed samples $\mathcal{Z}_p$ around x and the set of pseudo-labels $f(\mathcal{Z}_p) = \{f(z) \mid z \in \mathcal{Z}_p\}$. (i) x is converted into a binary vector $x' \in \{0, 1\}^{m'}$, (ii) perturbed samples are generated by drawing non-zero elements from x' uniformly at random, and (iii) the perturbed samples are converted back to the original space.
2. Learning a linear classifier g using $\mathcal{Z}_p$ and $f(\mathcal{Z}_p)$ by minimizing the following loss function:
$$\mathcal{L}(f, g, \pi_x) = \sum_{z \in \mathcal{Z}_p} \pi_x(z)(f(z) - g(z))^2, \tag{1}$$
where $\pi_x(z)$ is a weight function designed to be larger for samples closer to x, typically defined 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, it does not explicitly indicate the region for generating perturbed samples, making it difficult for users to assess the effective scope of the explanation [11].

Anchor [11] Anchor maximizes the coverage of a rectangular region containing the focal point x, expressed as a conjunction of feature predicates (a rule) as long as the probability of the black-box classifier f outputting $f(x)$ within the region exceeds a given threshold τ (Fig. 3(b)). It aims to highlight important features contributing significantly to the output. For a discrete m-dimensional input space $\mathbb{D}^m$, 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 predicate $a_i(z)$ evaluates to true ($= 1$) when $z_i = x_i$ and false ($= 0$) otherwise. The reliability of the explanation is defined as the "accuracy" of the rule, and the generality of the explanation is defined as the "coverage" of the rule. The accuracy $\mathrm{acc}(A)$ and coverage $\mathrm{cov}(A)$ of the rule A are defined as follows:

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

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

where $\mathcal{D}(z|A)$ is the conditional distribution in the region where the rule A returns true. $\mathrm{acc}(A)$ represents the probability that the output of f matches between the perturbation $z \sim \mathcal{D}(z|A)$ and the focal point x, and $\mathrm{cov}(A)$ expresses the probability that the perturbation z fits into A. Anchor maximizes coverage as long as the accuracy of the rule A exceeds a given threshold τ. However, eq. (2) is not directly computable. Introducing a confidence level $1 - \delta$ ($0 \leq \delta \leq 1$), the accuracy constraint is relaxed as follows:

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

Thus, the following optimization problem is solved:

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

3.2 Overview

We propose R-LIME, a method that aims to address the limitations of LIME [10] and Anchor [11]. Our method locally approximates the given black-box classifier f around the focal point x by a linear classifier g similar to LIME, but generates the perturbed samples from a rectangular region similar to Anchor so that the generality of approximation is explicitly provided (Fig. 3(c)).

Anchor maximizes the coverage of region A as long as the probability of the output of the black-box classifier f matching $f(x)$ within A exceeds a given threshold τ. R-LIME, on the other hand, learns a linear classifier g within the rectangular region A and maximizes the coverage of A under lower constraints on the accuracy of g. We modify Anchor's definition of accuracy in eq. (2) as follows:

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

where G is a hypothesis space of possible linear classifiers. By solving the optimization problem in eq. (5) under the modified definition of accuracy in eq. (6), we can select the rule that enables explanation with high accuracy and generality.

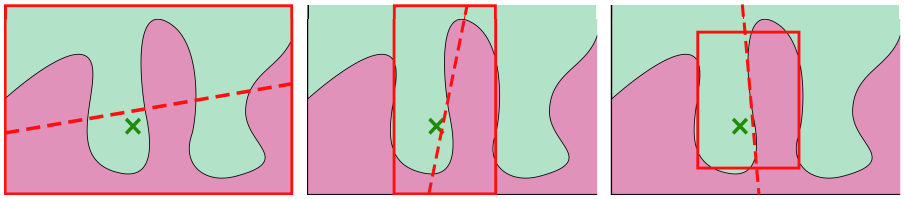


Fig. 4: Overview of the R-LIME algorithm. The progression of the algorithm is illustrated from left to right. The solid line represents the rectangular region A, and the dashed line represents the linear approximation model g learned within A. The initial value of A is an empty rule (entire input space), and predicates are added to A, reducing coverage. The process continues until $\mathrm{acc}(A) \geq \tau$ is satisfied, at which point the rule with the maximum coverage is output.

3.3 Algorithm

The algorithm of R-LIME is mainly based on that used in Anchor[11]. For non-convex optimization problems like eq. (5), greedy search are often used. But greedy methods tend to converge to local optima, so we use beam search, which selects multiple candidates at each iteration. The pseudo-code is shown in Algorithm 1.

Generating New Candidate Rules To generate new candidate rules, one additional predicate is added to each of the B candidate rules selected in the previous iteration. The pseudo-code is shown in Algorithm 2. $T(x)$ is the set of predicates $\{a_1, \ldots, a_m\}$, where $a_i(z)$ evaluates to true when $z_i = x_i$ and false otherwise. $T(x) \setminus A$ is the set of predicates in $T(x)$ not included in rule A.

Searching Rules with Highest Accuracy Given the set of candidate rules $\bar{\mathcal{A}}$, the algorithm selects the B candidate rules with the highest accuracy. This problem can be formulated as best arm identification in the multi-armed bandit framework. Each candidate rule $A_i \in \bar{\mathcal{A}}$ is considered as an arm, and reward of arm a_i follows a Bernoulli distribution with $P(X = 1) = \mathrm{acc}(A_i)$. 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 z and $f(z)$ after each trial. To efficiently search the rule (arm) with the highest accuracy, we employ the KL-LUCB algorithm [5]. The pseudo-code is shown in Algorithm 3. For tolerance $\epsilon \in [0, 1]$, the KL-LUCB algorithm guarantees below:

$$P(\min_{A \in \bar{\mathcal{A}}} \mathrm{acc}(A) \geq \min_{A' \in \mathcal{A}} \mathrm{acc}(A') - \epsilon) \geq 1 - \delta. \tag{7}$$

However, the KL-LUCB algorithm assumes that the reward distribution for each arm remains unchanged, while our method updates the classifier g_i with each sampling, which may not satisfy the assumption. This issue is discussed further in section 5.2.

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)

$A^* \leftarrow$ **null**, $\mathcal{A}_0 \leftarrow \emptyset$, $t \leftarrow 0$ ▷ Initialize the set of candidate rules $\mathcal{A}_0$ to $\emptyset$
while $A^* =$ **null do**
  $t \leftarrow t+1$
  $\bar{\mathcal{A}}_t \leftarrow$ GenerateCands$(\mathcal{A}_{t-1})$
  $\mathcal{A}_t \leftarrow$ B-BestCands$(\bar{\mathcal{A}}_t, \mathcal{D}, B, \epsilon, \delta)$
  $A^* \leftarrow$ LargestCand$(\mathcal{A}_t, \tau, \delta)$
end while

Algorithm 2 Generating new candidate rules

function GenerateCands$(\mathcal{A}, x)$
  if $\mathcal{A} = \emptyset$ **then return** $\{true\}$ ▷ An initial empty rule always returns *true*
  $\bar{\mathcal{A}} \leftarrow \emptyset$
  for all $A \in \mathcal{A}$ **do**
    for all $a \in (T(x) \setminus A)$ **do**
      $\bar{\mathcal{A}} \leftarrow \bar{\mathcal{A}} \cup (A \wedge a)$ ▷ Get a new rule by adding a new predicate a to A
    end for
  end for
  return $\bar{\mathcal{A}}$
end function

Algorithm 3 Searching rules with highest accuracy (KL-LUCB [5])

function B-BestCands$(\bar{\mathcal{A}}, \mathcal{D}, B, \epsilon, \delta)$
  initialize $\mathrm{acc}, \mathrm{acc}_u, \mathrm{acc}_l$ for $\forall A \in \bar{\mathcal{A}}$
  $\mathcal{A} \leftarrow$ B-ProvisionallyBestCands$(\bar{\mathcal{A}})$ ▷ B rules with highest accuracy
  $A \leftarrow \arg\min_{A \in \mathcal{A}} \mathrm{acc}_l(A, \delta)$ ▷ The rule with the smallest lower bound
  $A' \leftarrow \arg\max_{A' \notin (\bar{\mathcal{A}} \setminus \mathcal{A})} \mathrm{acc}_u(A', \delta)$ ▷ The rule with the largest upper bound
  while $\mathrm{acc}_u(A', \delta) - \mathrm{acc}_l(A, \delta) > \epsilon$ **do**
    sample $z \sim \mathcal{D}(z|A), z' \sim \mathcal{D}(z'|A')$
    update $\mathrm{acc}, \mathrm{acc}_u, \mathrm{acc}_l$ for A and A'
    $\mathcal{A} \leftarrow$ B-ProvisionallyBestCands$(\bar{\mathcal{A}})$
    $A \leftarrow \arg\min_{A \in \mathcal{A}} \mathrm{acc}_l(A, \delta)$
    $A' \leftarrow \arg\max_{A' \notin (\bar{\mathcal{A}} \setminus \mathcal{A})} \mathrm{acc}_u(A', \delta)$
  end while
  return $\mathcal{A}$
end function

Algorithm 4 Searching a rule with highest coverage under constraint

function LargestCand$(\mathcal{A}, \tau, \delta)$
  $A^* \leftarrow$ **null** ▷ If no rule satisfies the constraint, return **null**
  for all $A \in \mathcal{A}$ s.t. $\mathrm{acc}_l(A, \delta) > \tau$ **do**
    if $\mathrm{cov}(A) > \mathrm{cov}(A^*)$ **then** $A^* \leftarrow A$
  end for
  return A^*
end function

Searching a Rule with Highest Coverage under Constraint To satisfy the constraint imposed by eq. (4), a rule A needs to meet the following condition:

$$\mathrm{acc}_l(A, \delta) > \tau, \tag{8}$$

where $\mathrm{acc}_l(A, \delta)$ is the lower limit of the $100(1-\delta)\%$ confidence interval for $\mathrm{acc}(A)$. If the set of candidate rules $\mathcal{A}$ includes rules satisfying eq. (8), the one with the maximum coverage among them is selected, then the iteration is terminated. If $\mathcal{A}$ does not contain any rule satisfying eq. (8), it returns **null**, and proceeds to the next iteration. The pseudo-code is presented in Algorithm 4.

3.4 Computational Complexity

Post-hoc explanation methods including LIME, Anchor, and R-LIME need to sample a perturbation vector and get the output of the black-box model multiple times, which is computationally expensive. The number of samples required for LIME is $|\mathcal{Z}_p|$, which is the number of samples designated by the user. On the other hand, the expected number of samples required for Anchor and R-LIME is bounded by $\mathcal{O}[m \cdot \mathcal{O}_{\mathrm{MAB}[B \cdot m, B]}]$, where $\mathcal{O}_{\mathrm{MAB}[B \cdot m, B]}$ is the expected number of samples for best arm identification finding the best B arms from $B \cdot m$ arms. For KL-LUCB algorithm [5],

$$\mathcal{O}_{\mathrm{MAB}[B \cdot m, B]} = \mathcal{O}\left[\frac{Bm}{\epsilon^2} \log \frac{Bm}{\epsilon^2 \delta}\right]. \tag{9}$$

Then the total expected number of samples for Anchor and R-LIME is bounded by

$$\mathcal{O}\left[\frac{Bm^2}{\epsilon^2} \log \frac{Bm}{\epsilon^2 \delta}\right]. \tag{10}$$

For each iteration of KL-LUCB algorithm, R-LIME needs to update the linear classifier g_i, which is not required in Anchor. If we use logistic regression as the linear classifier and update it by stochastic gradient descent (SGD) [12], the computational complexity of updating g_i is $\mathcal{O}(m)$. It is negligible compared to the complexity of generating a perturbed sample, which is $\mathcal{O}(m^2)$ if we get a sample from a multivariate normal distribution using Cholesky decomposition in advance. Overall, the computational complexity of R-LIME is comparable to that of Anchor.

4 Experiments

To verify the effectiveness of the proposed method, we compared LIME and R-LIME using a real-world dataset.

Table 1: Attributes of the recidivism dataset used in the experiments. Continuous features are all 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

4.1 Qualitative Evaluation

Experimental Setup We used the recidivism dataset [14] for our experiments. The dataset contains personal information on 9549 prisoners released from North Carolina prisons between July 1, 1979, and June 30, 1980. As shown in Table 1, the dataset includes 19 items such as race (*Race*), gender (*Gender*), presence of alcohol dependence (*Alcohol*), number of prior offenses (*Priors*), and presence of recidivism (*Recidivism*). For this experiment, we treated the binary classification problem of predicting the presence of recidivism (*Recidivism*) as the target label. We discretized continuous features and removed missing values, resulting in 15 features.

We splitted the dataset into training data (7639 instances) and test data (955 instances), and trained a random forest model with 50 trees as the black-box classifier using the training data. Then, LIME and R-LIME explanations were generated for two instances extracted from the test data (Fig. 5). For R-LIME, we used logistic regression as the linear approximation model, and a multivariate normal distribution estimated from the training data as the distribution $\mathcal{D}$. 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 $\epsilon = 0.05$. The accuracy threshold τ was set to $\tau = 0.70, 0.80, 0.90$.

This problem setting can be considered as a case where a complex machine learning 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.

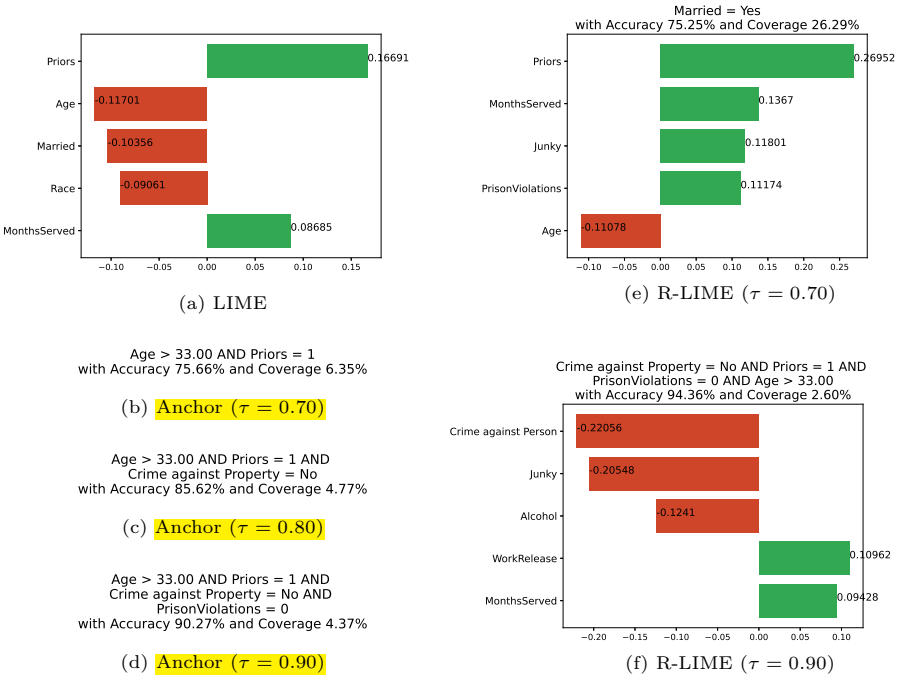
Race	Black (0)
Alcohol	No (0)
Junky	No (0)
Supervised Release	Yes (1)
Married	Yes (1)
Felony	No (0)
WorkRelease	Yes (1)
Crime against Property	No (0)
Crime against Person	No (0)
Gender	Male (1)
Priors	1
YearsSchool	8.00 < YearsSchool <= 10.00 (1)
PrisonViolations	0
Age	Age > 33.00 (3)
MonthsServed	4.00 < 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. 5: Two instances sampled from training data of recidivism dataset. Each number in parentheses represents the integer value assigned to the corresponding categorical value.



(a) LIME

(e) R-LIME ($\tau = 0.70$)

(b) Anchor ($\tau = 0.70$)

(c) Anchor ($\tau = 0.80$)

(d) Anchor ($\tau = 0.90$)

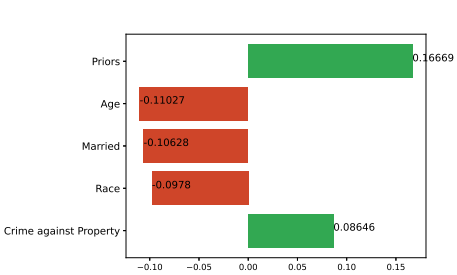
(f) R-LIME ($\tau = 0.90$)

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

Experimental Results The results of the experiment are shown in Figs. 6 and 7. The values assigned to each feature name represent the contribution (weight of the linear classifier) to the output of the black-box classifier, normalized such that the absolute sum is 1. The figures display the 5 features with the highest absolute contribution.

Explanations generated by LIME (Figs. 6(a) and 7(a)) provide insights that having a prior offenses (*Priors*), being served for a long time in prison (*MonthsServed*), and committing a crime against property (*Crime against Property*) primarily contribute to the positive prediction (prediction that the prisoner will be re-arrested). On the other hand, being elderly (*Age*), being married (*Married*), and being of white race (*Race*) contribute to the negative prediction (prediction that the prisoner will not be re-arrested). While these LIME explanations provide valuable insights into the behavior of the black-box model, they do not explicitly indicate the application scope of the explanations, leaving users unable to determine to which prisoners the explanations are applicable.

In contrast, R-LIME expresses the application scope of explanations as a conjunction of feature predicates. For example, the explanation for instance A under $\tau = 0.70$ (Fig. 6(e)) indicates that it is applicable only to married prisoner (*Married* = *Yes*). R-LIME explanations also provide their accuracy and coverage, allowing users to evaluate reliability and generality of the explanations. For ex-



(a) LIME

Priors = 1 AND PrisonViolations = 0
with Accuracy 72.03% and Coverage 26.71%

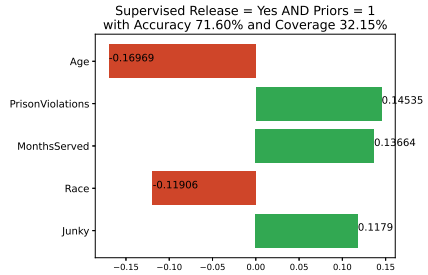
(b) Anchor ($\tau = 0.70$)

Priors = 1 AND PrisonViolations = 0 AND
YearsSchool > 11.00 AND
4.00 < MonthsServed <= 9.00
with Accuracy 83.11% and Coverage 2.34%

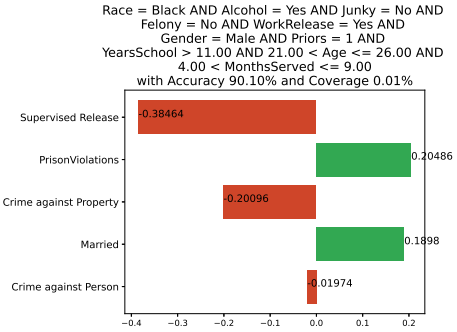
(c) Anchor ($\tau = 0.80$)

YearsSchool > 11.00 AND Priors = 1 AND
PrisonViolations = 0 AND
4.00 < MonthsServed <= 9.00 AND Junky = No AND
Married = No AND Gender = Male AND
WorkRelease = Yes AND Alcohol = Yes AND
Race = Black
with Accuracy 93.13% and Coverage 0.09%

(d) Anchor ($\tau = 0.90$)



(e) R-LIME ($\tau = 0.70$)



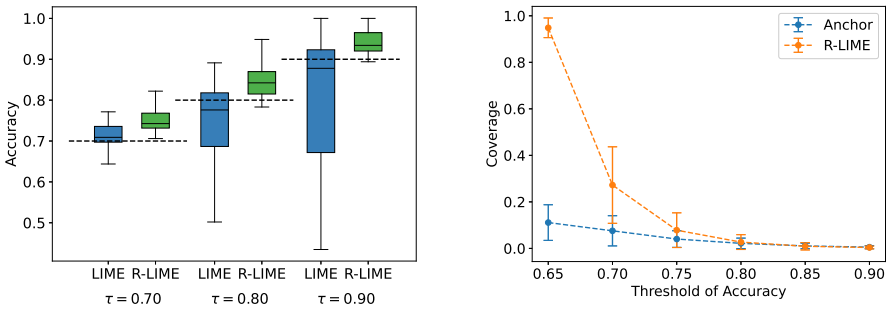
(f) R-LIME ($\tau = 0.90$)

Fig. 7: Explanation for Instance B by LIME and R-LIME.

ample, the coverage of the explanation for instance B under $\tau = 0.90$ (Fig. 7(f)) is 0.01%, indicating that the decision boundaries around instance B are complex, making it challenging to obtain a high-accuracy linear approximation. This information allows users to discern that the application scope of this explanation is very narrow, limiting its utility.

4.2 Quantitative Evaluation: LIME vs. R-LIME

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



(a) Comparison of local accuracy between LIME and R-LIME. R-LIME achieved higher and less variable accuracy compared to LIME.

(b) Comparison of coverage between Anchor and R-LIME. R-LIME achieved higher coverage compared to Anchor for almost values of τ, especially for relatively small τ.

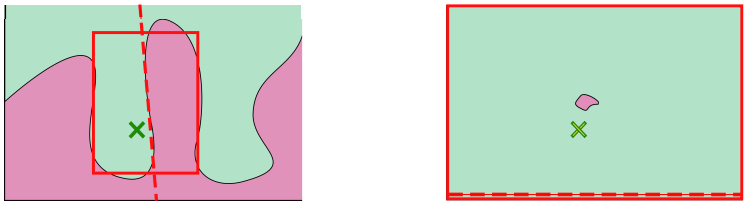
Fig. 8: Comparison of existing methods (LIME, Anchor) and R-LIME.

Experimental Results The results are presented in Fig. 8(a), showing the distribution of the local accuracy of the linear classifiers learned by LIME and R-LIME. R-LIME achieved higher accuracy compared to LIME for all values of τ. This suggests that the linear classifiers learned by LIME and R-LIME differ significantly, and R-LIME learns a high-accuracy linear classifier adapted to the optimized rectangular region. Additionally, as τ increases, the variability in the accuracy of LIME widens. This indicates that the linear classifiers learned by LIME may not function effectively as approximation models depending on how the region is selected.

4.3 Quantitative Evaluation: Anchor vs. R-LIME

Experimental Setup To demonstrate that R-LIME explanations are more general than Anchor, we conducted a comparison of the coverage of explanations between Anchor and R-LIME. Under the same settings as in section 4.1, we generated Anchor and R-LIME explanations for 100 instances from the test data of the recidivism dataset, under the values of $\tau = 0.70, 0.80, 0.90$.

Experimental Results The results are presented in Fig. 8(b), showing the coverage of the explanations by Anchor and R-LIME. The coverage of explanations generated by R-LIME is higher compared to Anchor for almost values of τ, especially for relatively small τ. It is because of the flexibility of the linear approximation models learned by R-LIME, which captures the decision boundary more precisely. In contrast, Anchor uses only the intervals of each feature discretized in advance, which cannot capture the decision boundary flexibly and makes its scope narrow.



(a) R-LIME for balanced label distribution. (b) R-LIME for imbalanced label distribution.

Fig. 9: Behavior of R-LIME for balanced and imbalanced label distribution. In case of imbalanced label distribution, the approximation region covers the entire input space and the linear approximation model always outputs the majority label.

Table 2: Deviation between the estimated accuracy and the true accuracy of the linear classifier learned by R-LIME. The deviation 0.012 ± 0.017 was relatively small considering the confidence level $1 - \delta = 0.95$.

	Estimated acc.	True acc.	Deviation
Average	.811	.829	.012
Standard Deviation	.018	.023	.017

5 Discussion

5.1 Behavior Regarding Imbalanced Label Distribution

R-LIME may generate less useful explanations when the distribution of black-box classifier outputs is imbalanced. When the ratio of outputting the minority label is less than $1 - \tau$, where τ is the accuracy threshold, the approximation region generated by R-LIME covers the entire input space, and the learned linear classifier always outputs the majority label (Fig. 9).

A first possible solution to this problem is modifying the loss function. Using weighted logistic loss or Focal Loss [6] 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

$$\left(\mathbb{E}_{z \sim \mathcal{D}(z|A)}[\mathbb{1}_{f(z)=1}] - \frac{1}{2}\right)^2 < \mu \tag{11}$$

could suppress the excessive expansion of the approximation region.

5.2 Changes in Reward Distribution in Best Arm Identification

For R-LIME, the problem of selecting the rule with the highest accuracy is formulated as the best arm identification problem in multi-armed bandit framework, and solved using the KL-LUCB algorithm [5]. However, this algorithm

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

We conducted an experiment to evaluate the deviation between the estimated accuracy and the true accuracy. We generated explanations for 3200 data instances sampled from the dataset, and compared the estimated accuracy with the true accuracy. The true accuracy was calculated based on 1000 instances sampled within the approximation region. The results in Table 2 show a mean deviation of 0.012 with a standard deviation of 0.017. By considering the confidence level $1 - \delta = 0.95$, the deviation was relatively small. While there are concerns about the theoretical validity of using the KL-LUCB algorithm, our results suggest that the deviation is not significant in practice.

6 Conclusion

Existing methods for local model-agnostic explanations of black-box classifiers, such as LIME and Anchor, have limitations that they cannot achieve interpretability of both the explanation and its application scope. To address these challenges, we proposed R-LIME, a method that locally and linearly approximates the decision boundary of a black-box classifier and provides a rectangular approximation region, which is interpretable for users due to being expressed as a conjunction of feature predicates. We proposed an algorithm to maximize coverage of the approximation region as long as the accuracy of the linear approximation model exceeds a given threshold. Comparing the outputs of LIME and R-LIME on the real-world dataset, we demonstrated that R-LIME provides a clear application scope of the explanation, can be evaluated by users for its reliability and generality, and achieves higher and less variable local accuracy compared to LIME. Finally, we discussed the instability of behavior against imbalanced label distributions and raised questions about the theoretical validity of using the KL-LUCB algorithm for our problem.

References

1. Apley, D.W., Zhu, J.: Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. Journal of the Royal Statistical Society Series B: Statistical Methodology **82**(4), 1059–1086 (2020). https://doi.org/10.1111/rssb.12377, https://doi.org/10.1111/rssb.12377
2. Bach, Sebastian AND Binder, Alexander AND Montavon, Grégoire AND Klauschen, Frederick AND Müller, Klaus-Robert AND Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PLOS ONE **10**(7), 1–46 (2015). https://doi.org/10.1371/journal.pone.0130140, https://doi.org/10.1371/journal.pone.0130140

3. Friedman, J.H.: Greedy function approximation: A gradient boosting machine. The Annals of Statistics **29**(5), 1189–1232 (2001), http://www.jstor.org/stable/2699986
4. Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., Giannotti, F.: Local rule-based explanations of black box decision systems (2018)
5. 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
6. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, 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
7. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)
8. Montavon, G., Lapuschkin, S., Binder, A., Samek, W., Müller, K.R.: Explaining nonlinear classification decisions with deep taylor decomposition. Pattern Recognition **65**, 211–222 (2017). https://doi.org/https://doi.org/10.1016/j.patcog.2016.11.008, https://www.sciencedirect.com/science/article/pii/S0031320316303582
9. Radulovic, N., Bifet, A., Suchanek, F.: BELLA: Black box model explanations by local linear approximations (2023)
10. 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
11. 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
12. Robbins, H., Monro, S.: A stochastic approximation method. The Annals of Mathematical Statistics **22**(3), 400–407 (1951), http://www.jstor.org/stable/2236626
13. Samek, W., Montavon, G., Lapuschkin, S., Anders, C.J., Müller, K.R.: Explaining deep neural networks and beyond: A review of methods and applications. Proceedings of the IEEE **109**(3), 247–278 (2021). https://doi.org/10.1109/JPROC.2021.3060483
14. Schmidt, P., Witte, A.D.: Predicting Recidivism in North Carolina, 1978 and 1980. Inter-university Consortium for Political and Social Research (1988)