applied sciences

MDPI

Article

gbt-HIPS: Explaining the Classifications of Gradient Boosted Tree Ensembles

Julian Hatwell *, Mohamed Medhat Gaber and R. Muhammad Atif Azad

Data Analytics and Artificial Intelligence Research Group, Faculty of Computing, Engineering and the Built Environment, Birmingham City University, Curzon Street, Birmingham B5 5JU, UK; mohamed.gaber@bcu.ac.uk (M.M.G.); atif.azad@bcu.ac.uk (R.M.A.A.)

* Correspondence: julian.hatwell@bcu.ac.uk

Abstract: This research presents *Gradient Boosted Tree High Importance Path Snippets* (gbt-HIPS), a novel, heuristic method for explaining gradient boosted tree (GBT) classification models by extracting a single classification rule (CR) from the ensemble of decision trees that make up the GBT model. This CR contains the most statistically important boundary values of the input space as antecedent terms. The CR represents a hyper-rectangle of the input space inside which the GBT model is, very reliably, classifying all instances with the same class label as the explanandum instance. In a benchmark test using nine data sets and five competing state-of-the-art methods, gbt-HIPS offered the best trade-off between coverage (0.16–0.75) and precision (0.85–0.98). Unlike competing methods, gbt-HIPS is also demonstrably guarded against under- and over-fitting. A further distinguishing feature of our method is that, unlike much prior work, our explanations also provide counterfactual detail in accordance with widely accepted recommendations for what makes a good explanation.

Keywords: explainable artificial intelligence; human-understandable AI systems; gradient boosting; black box problem; machine learning interpretability

Citation: Hatwell, J.; Gaber, M.M.; Azad, R.M.A. gbt-HIPS: Explaining the Classifications of Gradient Boosted Tree Ensembles. *Appl. Sci.* **2021**, *11*, 2511. https://doi.org/10.3390/app11062511

Academic Editor: Federico Divina

Received: 30 January 2021
Accepted: 6 March 2021
Published: 11 March 2021

1. Introduction

Gradient boosted tree (GBT) models [1] remain the state-of-the-art for many "shallow" learning tasks that are based on structured, tabular data sets [2–4]. Such tasks are often still found in high-stakes decision making domains, such as medical decision making [5–8]; justice and law [9,10]; financial services [11–13]; and defence and military intelligence [14]. In these and similar domains, there is a high burden of accountability for decision makers to explain the reasoning behind their decisions. This burden only increases with the introduction of machine learning (ML) into decision making processes [15]. So, the very high accuracy and ease of use of GBT models is not enough to encourage their adoption because GBT models also typify the "black box" problem of uninterpretability. Hence, research in interpretable machine learning (IML) and explainable artificial intelligence (XAI) has emerged to overcome these barriers to adoption.

Deriving explanations from the complex structure of GBT models (as an ensemble of decision trees) has remained an open challenge. Gradient-based attribution methods that are used to explain deep learning (DL) models and neural networks are unsuitable here because the internal sub-units of a GBT model are non-parametric and non-differentiable decision nodes. The available IML and XAI methods have several disadvantages.

IML methods can be used to facilitate the interpretation of a GBT model, as well as other types of decision tree ensemble, also known as decision forests (DFs). These methods generate a cascading rule list (CRL) as an inherently interpretable proxy model. First, a very large set of candidate classification rules (CRs) is generated. The defragTrees [16] and inTrees [17] methods achieve this by extracting all possible CRs from the decision trees in the DF. Bayesian rule lists (BRLs) [18] use a different approach, which is to mine the rules directly from the training data. For all three methods, the candidate set of CRs is

then merged or pruned using, e.g., some Bayesian formulation into the final CRL. One disadvantage with this approach is the very high computational and memory cost of exploring the combinatorial search space of candidate CRs. Another is imperfect fidelity, which means that the proxy does not always agree with the reference model's classification. This is a consequence of the proxy's simplification of the complex behaviour of the black box reference model. Disagreement between the two results in a failure to explain the reference model.

Currently available XAI methods for explaining GBT are the so called model-agnostic methods. Methods include locally interpretable model-agnostic explanations (LIMEs) [19], Shapley additive explanations (SHAP) [20], local rule-based explanations (LORE) [21], and anchors [22]. These general purpose methods are said to explain any model. This flexibility is achieved by using a synthetic data set to probe the reference black box model and infer a relationship between its inputs and outputs. However, these methods are disadvantaged because there is no introspection of the reference model or target distribution, which is thought to be essential for reliable explanations [23,24]. Furthermore, the explanations can exhibit variance because of the non-deterministic data generation [25,26]. High variance results in dissimilar explanations for similar instances. They can also be fooled by adversarial examples [27]. Furthermore, the synthetic data place too much weight on rare and impossible instances [28–30].

Recent contributions [31,32] present model-specific XAI approaches that probe DF internals and generate a single CR as an explanation. Thus, these newer methods overcome the combined disadvantages of prior work and have been shown to outperform the model-agnostic XAI methods and CRL-based IML methods on other classes of DF. Namely, the random forest [33] and AdaBoost [34] models. Hence, there still remains a gap in the literature for model-specific explanation methods that target GBT models. This research addresses that gap.

In this paper, we present Gradient Boosted Tree High Importance Path Snippets (gbt-HIPS), a novel, model-specific algorithm for explaining GBT models. The method is validated against the state-of-the-art in a comprehensive experimental study, using nine data sets and five competing methods. gbt-HIPS offered the best trade off between coverage (0.16–0.75) and precision (0.85–0.98) and was demonstrably guarded against under- and over-fitting. A further distinguishing feature of our method is that, unlike much prior work, our explanations also provide counterfactual detail in accordance with widely accepted best practice for good explanations.

The remainder of this paper is set out as follows: Section 2 presents gbt-HIPS, our novel algorithm for generating explanations of GBT classification; Section 3 describes our experimental procedures; Section 4 discusses the results; and Section 5 concludes the paper with some ideas for future research.

2. The Gbt-HIPS Method

This section presents Gradient Boosted Trees High Importance Path Snippets (gbt-HIPS) method in detail. Each step is illustrated in the conceptual diagram in Figure 1 and detailed in the following sections.

The design of gbt-HIPS takes into consideration Miller's key principles [35]. To the best of our knowledge, very few prior works have responded to those insights so directly, with the exception of [31,32]. The "model of self" principle suggests that, in order to be a true representation of the model's internal logic, the explanation must, in some way, be formed from the model internals. This idea is echoed in the four axioms given in [23], e.g., "Explanation without introspection is not explanation." The form of explanation is a single CR extended with counterfactual detail, as is expected by Miller's "contrastive" principle. The CR form naturally aligns with the other key principles of "selectivity" and "minimal completeness." These are satisfied when the rule contains the right combination of antecedent terms to cover a non-trivial volume of the input space, while each individual term is necessary to ensure maximal rule precision. The counterfactual detail is the loss of

precision that arises when any single antecedent term is violated. This fuzzy counterfactual is a necessary adaptation for data that contains any continuous variables and, in fact, provides much more information than a discrete change of class label. For full details, refer to [31].

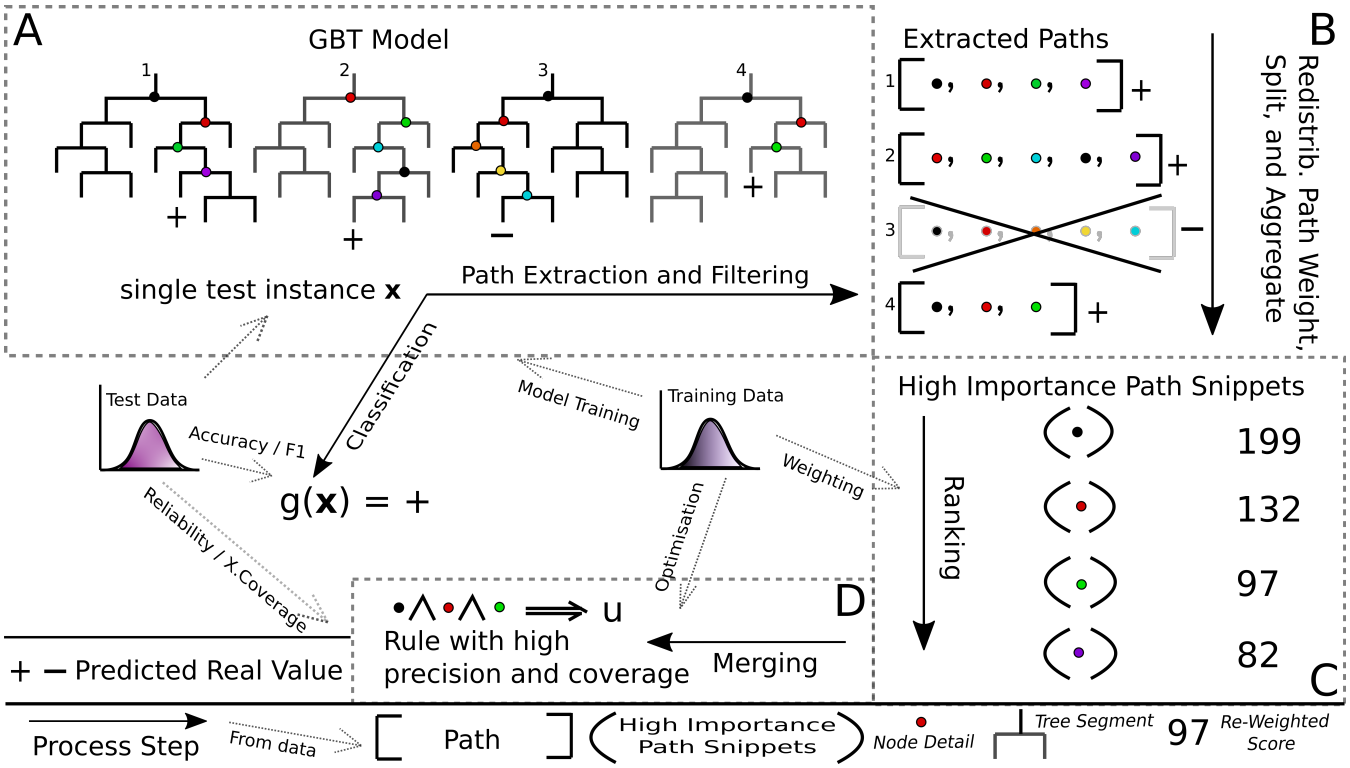

Figure 1. Conceptual diagram of Gradient Boosted Trees High Importance Path Snippets (gbt-HIPS). There are four steps: (**A**) path extraction and filtering, (**B**) decompose the paths and redistribute the predicted logit values among the nodes, (**C**) scoring and ranking, (**D**) merging and pruning.

The gbt-HIPS algorithm follows a greedy, breadth-first, heuristic search. The first step is to set the CR consequent as the black box model's output. Thus, by design, the CR will always agree with the black box for the explanandum. Then, candidate decision nodes are extracted from the decision trees in the ensemble with two filtering steps unique to this research. The importance of each decision node (and therefore its opportunity to be included in the rule antecedent) is calculated by means of a statistically motivated procedure, based on relative entropy. These weighted decision nodes are referred to as path snippets. The resulting snippets are merged into a final rule according to a simple, greedy heuristic. The following paragraphs describe the process in full detail.

2.1. Path Extraction and Filtering

The first step is to extract the decision path of the explanandum instance **x** from every decision tree in the GBT model g. This design choice means that, unlike in IML methods, the rest of the model is ignored when generating the explanation for the given classification event $g(\mathbf{x})$. This filtering reduces the size of the search logarithmically and is justified because there is only one possible path for **x** down each tree. So, none of the other paths contribute to the final output.

In the multi-class case, GBT models consist of K one-vs-all binary logistic classifiers. So, classification is normally modified such that the winning class is determined by the kth classifier that has the largest positive value. gbt-HIPS uses only paths from this kth classifier.

Path extraction simply records the detail at each decision node as it is traversed by the explanandum instance on its way to a terminal node. The decision path consists of this set of decision nodes along with the real-valued output from the terminal node. Recall

that GBT uses regression trees, whose aggregated outputs make a log odds prediction. The extracted paths are then filtered to retain only the paths whose terminal node output has the same sign as the ensemble output (always positive for multi-class settings but could be either sign in binary settings). This stage of filtering is justified because those paths that do not agree with the overall ensemble "lose the election." The decision nodes in the retained paths contain all the information about the model's output. The excluded paths are expected to capture noise, or perhaps attributes that are more strongly associated with the alternative class.

2.2. Redistribute Path Weight, Split the Paths, and Aggregate the Nodes

This second step is critical for assigning importance scores to the decision nodes. The path's weight is the absolute value returned by the path's terminal node. The path weights represent each individual tree's contribution to the log odds prediction and must be fairly distributed over each decision node in the path. The redistribution must take into account the node order in the originating path as well as the predictive power of the node itself. The KL-divergence, also known as relative entropy, is ideal for this purpose because it measures information gained if a new distribution (P) is used, instead of a reference distribution (P'). The KL-divergence is calculated as follows:

$$D_{KL}(P \parallel P') = -\sum_{k=1}^{K} P \log\left(\frac{P}{P'}\right) \tag{1}$$

where P is the distribution of class labels for instances that reach a given node in the path and P' is the distribution that reaches the previous node in the path. In the case of the root node, which is first in the path, P' is simply the prior distribution. Here, the quantities are estimated using the training set or any other large i.i.d. sample. Once the relative entropy for the last decision node in the path is evaluated, the values for the entire path are normalised, such that their total is equal to that of the path weight.

The paths are then disaggregated into the individual, weighted decision nodes and stored in a key-value dictionary, with the decision nodes as keys and the redistributed path weights as values. The weights for identical nodes are aggregated by summation as they enter the dictionary. While this operation is straightforward for nodes that represent discrete or categorical features, there is a complication with nodes that act on continuous features that is a natural consequence of GBT training. On each training iteration, the instances are re-weighted, which alters the target distribution. See [1] for further details. A further perturbation occurs in the very common stochastic GBT variant. Stochastic GBT takes random samples of training instances on each iteration, in addition to the aforementioned re-weighting. Hence, when a new decision tree is induced over the modified distribution, the exact location at which continuous features might be partitioned can move from one iteration to the next in a non-deterministic manner. That is to say, many unique values may all represent the same decision boundary. Competing methods (such as LIME, Anchors, SHAP and the IML models) avoid this problem by preprocessing the continuous features into arbitrary, quartile bins. There are several problems with this approach, not least of which is a loss of information because the quartile bin medians are very unlikely to align with the optimal boundary values. gbt-HIPS instead delegates this discretisation step to the GBT model training. More specifically, it is the information theoretic step of decision tree induction that determines the location of each split in the continuous variables. gbt-HIPS applies a simple binning function to all the extracted decision node boundary values for each continuous feature.

This idea is illustrated in Figure 2. In this toy example, the data set shown has one feature, $x1$ and two classes. The negative class $-$ (blue) is found mostly around the middle of the range of $x1$ and is completely enclosed by the positive class $+$ (grey). A GBT model is trained for 1000 iterations using decision trees with $maxdepth = 2$, allowing each tree to find two boundary values. The effect of having a freshly perturbed training distribution

on each iteration is that each decision node represents a sample from the approximate locations of the optimal boundary values. The resulting decision node boundary values are binned with a simple histogram function. Thus, it is possible to find a very small number of near optimal boundary values to include in the explanations, removing the aforementioned unique value problem. It is clear from Figure 2 that the decision boundary bin medians of the model (shown in the lower panel) align very closely with the cluster boundaries. On the other hand, quartile bin medians that sit half-way between the quartile boundaries (superimposed on the kernel density plot, top panel) tend to split clusters in half or simply appear at arbitrary locations relative to the distribution. This simple example demonstrates that preprocessing by quartile binning does not lead to optimal candidate boundary values for the explanations while gbt-HIPS's post-processing binning does.

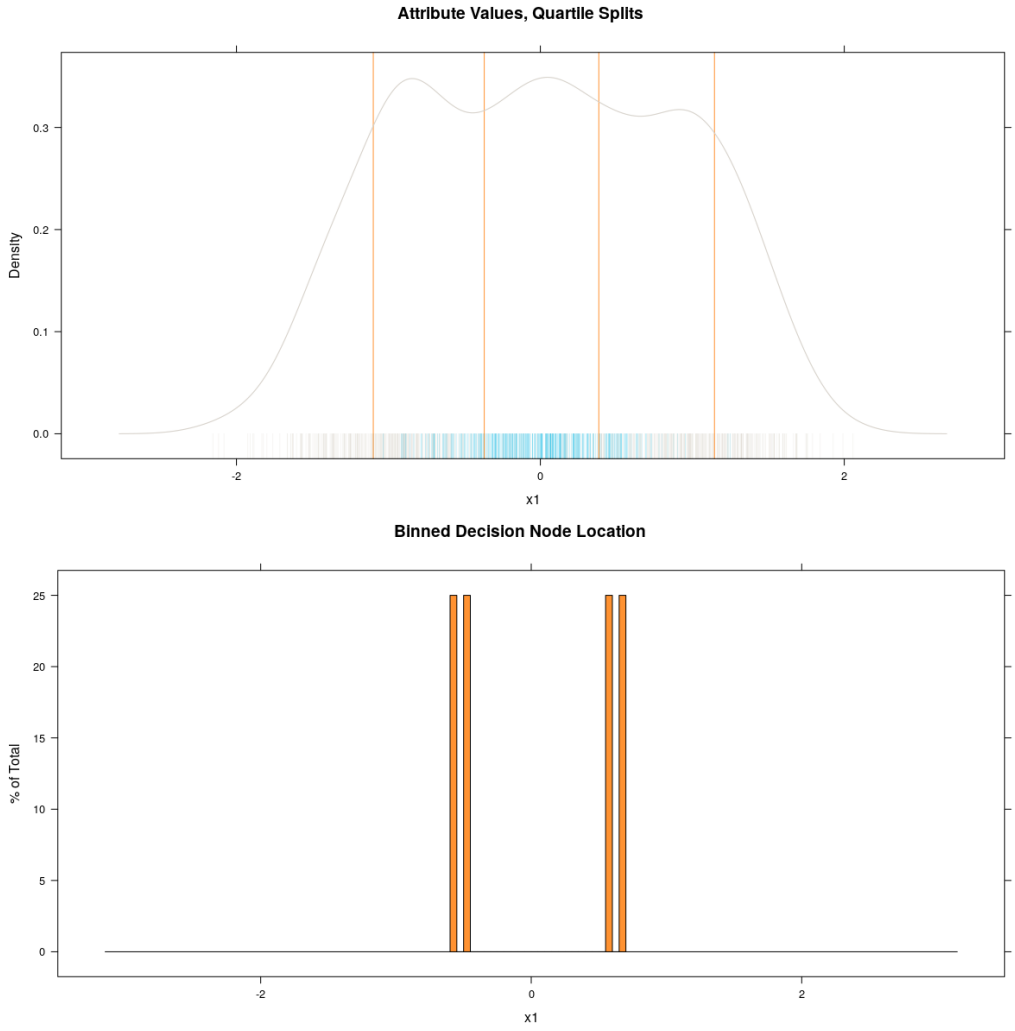

Figure 2. Comparison of preprocessing by quartile binning and postprocessing decision node binning for two non-linearly separable classes.

2.3. Ranking

At this point, the individual decision nodes are separated from their originating path and each decision has an aggregated weight. We refer to their new state as path snippets. The dictionary of unique path snippets created in the previous step is simply sorted by weight, in descending order. Ranking is essential for avoiding an exhaustive search in the next step, because the ordering controls the path snippets' opportunity to be included in the candidate explanation. It is reasonable at this point to filter out the path snippets with

the smallest aggregated weight using a top n or top n% hyper-parameter, as it will shorten the step that follows.

2.4. Merging and Pruning

The final step generates the CR-based explanation using a breadth first, greedy, heuristic search of the path snippets. Before the search commences, the first step is to set the rule consequent as the GBT model's classification of the explanandum instance $\mathbf{x}$. This step guarantees local accuracy. The search then begins from $\varnothing \implies g(\mathbf{x})$, the "null" rule that has an empty antecedent and maximum coverage. The first path snippet, at the top of the sorted dictionary, is appended to the rule's antecedent. The reliability on the training data is evaluated and this value stored. Then, one at a time in turn, path snippets from the sorted dictionary are added as candidate antecedent terms. If the union improves the reliability, the snippet is retained in the rule. If not, the snippet is simply discarded. In both cases snippets are removed from the dictionary.

To further reduce the number of iterations, any path snippets are deleted from the ranked list if they contain boundary values of continuous features that fall outside the current coverage region. That is, coverage is forced to decrease monotonically. These steps, merging a path snippet and pruning the dictionary, iterate until a target reliability threshold is met or the list is exhausted, as illustrated in Algorithm 1.

Algorithm 1 Rule Merging

procedure RULEMERGE(g, $\mathbf{X}$, $\mathbf{x}$, *dictionary*, ρ)
▷ Inputs: model, training data, explanandum, sorted map, target reliability.
  $\mathcal{C} \leftarrow \varnothing$ ▷ Candidate set of antecedent terms
  $r \leftarrow \zeta(g, \mathbf{X}, \mathbf{x})$ ▷ Prior reliability.
  while $r < \rho$ and length *dictionary* > 0 **do**
    $\mathbf{Z} \leftarrow \mathbf{X} \cap (\mathcal{C} \cup dictionary^{(1)})$
    ▷ Append top ranking path snippet. Partition only covered instances.
    if $\zeta(g, \mathbf{Z}, \mathbf{x}) > r$ **then**
      $\mathcal{C} \leftarrow \mathcal{C} \cup dictionary^{(1)}$ ▷ Path snippet added to rule.
      $r \leftarrow \zeta(g, \mathbf{Z}, \mathbf{x})$
    $dictionary \leftarrow dictionary \setminus dictionary^{(1)}$
    ▷ Remove top snippet. Reset indices.
  return $\mathcal{C}$

After rule merging completes, the candidate set of antecedent terms $\mathcal{C}$ is returned, forming the final candidate rule. This candidate is pruned of extraneous terms in a process that also generates the counterfactual detail, while enforcing minimal completeness, as required by Miller's principles of explanation [35]. The inclusion of extraneous terms can occur because the greedy heuristic only enforces a monotonic increase in reliability. Thus, terms that increase performance only very slightly may be included. Furthermore, some antecedent terms are rendered redundant through interaction with terms that are added subsequently. These non-optimal side-effects are to be expected with greedy, heuristic algorithms. Therefore, the pruning step iterates through the "point changes", removing any antecedent terms that are found to be extraneous. A point change is defined as the reversal of the inequality constraint of a single antecedent term. Therefore, point changes represent a set of "adjacent spaces" to that hyper-cube (or half-space) of the input space that is covered by the rule. Adjacent spaces are outside the hyper-cube by one axis-aligned step across a single rule boundary. To determine whether an antecedent term is extraneous, the reliability is evaluated on the training instances covered by each adjacent space. If reliability decreases by $<\delta$ (a user-defined parameter) inside an adjacent space, that antecedent term can be removed from the rule. The result is a shorter rule that has a greater coverage and whose reliability lies within the user-defined tolerance of the unpruned candidate's reliability.

2.5. Output

The final output is the rule, together with estimates of any requested statistics evaluated for the training set or other i.i.d. sample. Estimates of precision for each of the adjacent spaces convey the counterfactual detail. This formulation should aid the end user in validating the importance of each antecedent term.

An example of this output is given in Table 1 and is taken from the adult data set that is freely available from the UCI Machine Learning Repository [36]. In this classification task, models are trained to predict whether an individual has an annual income greater than or less than/equal to US$50 K using a set of input features related to demographics and personal financial situation. The explanandum instance here was selected at random from a held out test set. The GBT model classified this instance as having an income less than or equal to $50 K per annum. The explanation column shows the final CR, one row per term. This CR covers 53.8% of training samples with a precision of 98.8% (instances in the half-space that correctly receive the same classification). These boundary values include only the two attributes: (log) capital gain is less than 8.67 and marital status is not equal to married-civ, giving a very short rule that is trivial for human interpretation. The contrast column contains the counterfactual detail, which is the change in the rule's precision when the inequality in each antecedent term is reversed, one at a time, i.e., substituting $\leq$ for $>$, or $\neq$ for $=$. Reversing either one of these boundary values in this way (thus exploring the input space outside the enclosed half-space) creates a CR with either the opposite outcome or a result that is worse than a random guess if controlling for the prior distribution. This is, therefore, a very high quality explanation.

Table 1. An example of gbt-HIPS output.

Data Set:	Decision:	Explanation:	Contrast:	Confidence:
adult	Income $\leq$ \$50 K	lcapitalgain $\leq$ 8.67 $\wedge$ marital status $\neq$ married-civ	−85.0%−39.4%	Coverage 53.8% Precision 98.8%

3. Materials and Methods

The work described in the coming sections is reproducible using code examples in our github repository https://tinyurl.com/yxuhfh4e (5 March 2021).

3.1. Experimental Design

The experiments were conducted using both Python 3.6.x and R 3.5.x environments, depending on the availability of open-source packages for the benchmark methods. The hardware used was a TUXEDO Book XP1610 Ultra Mobile Workstation with Intel Core i7-9750H @ 2.60–4.50 GHz and 64GB RAM using the Ubuntu 18.04 LTS operating system.

This paper follows exactly the experimental procedures described in [31,32], which adopt a functionally grounded evaluation [37]. The use of this category is well justified because the present research is a novel method in its early stages, and there is already good evidence from prior human-centric studies demonstrating that end users prefer high precision and coverage CR-based explanations over additive feature attribution method (AFAM) explanations [21,22]. The efficacy of CR-based explanations is also already well-established by IML models [7,24,38,39]. Functionally grounded studies encourage large-scale experiments. So, this research will compare the performance of gbt-HIPS with five state-of-the-art methods on nine data sets from high-stakes decision-making domains.

The aforementioned precedents [21,22] measure the mean precision and coverage for the CR-based explanations generated from a held out set of instances that were not used in model training. Those precedents found that coverage and precision were effective as proxies to determine whether a human user would be able to answer the fundamental questions: "does a given explanation apply to a given instance?" and "with what confidence can the explanation be taken as valid?"

The experimental method uses leave-one-out (LOO) evaluation on held out data to generate a very large number of test units (explanations) in an unbiased manner from each

data set. This approach is better suited to the XAI setting because every explanation is independent of the training set, and independent of the set used to evaluate the statistics of interest. Each data set was large enough that any inconsistencies in the remaining $N - 1$ evaluation set were ignorable.

Aforementioned related work indicates that individual explanations can take between a fraction of a second and a few minutes to generate. This timing was confirmed in a pilot study, prior to the forthcoming experimental research. To balance the need for a large number of explanations against the time required to run all the tests, the experimental study will generate 1000 explanations or the entire test set, whichever number is smaller.

3.2. Comparison Methods and Data Sets

gbt-HIPS produces CR-based explanations. Direct comparisons are possible against other methods that either output a single CR as an explanation, or a rule list from which a single CR can be extracted. Readers that have some familiarity with XAI may question the omission of LIME [19] and SHAP [20] from this study since they are two of the most discussed explanation methods to date. However, as the authors of [20] make clear, these are AFAM and, therefore, of an entirely different class. There is no straightforward way to compare explanations from different classes as prior works have demonstrated [21,22]. For example, there is no way to measure the coverage of an AFAM explanation over a test set, whereas for a CR the coverage is unambiguous. Fortunately, Anchors [22] has been developed by the same research group that contributed LIME. Anchors can be viewed as a CR-based extension of LIME and its inclusion into this study provides a useful comparison to AFAM research. In addition to Anchors, LORE [21] is included, as another per-instance, CR-based explanation method. These are the only such methods that are freely available as open-source libraries for Python and R development environments.

We also included three leading CRL-based interpretable machine learning (IML) methods into the study design. When using CRL models, the first covering (or firing) rule is used to classify the instance and, thus, is also the stand-alone explanation. If there is no covering rule, the default rule is fired. This *null* rule simply classifies using the prior class majority. For the purposes of measuring rule length, a firing null rule has a length of zero. All selected methods are detailed in Table 2.

Table 2. Methods used in the experiments.

Method	IML/Local	Ref.
inTrees	IML	[17]
defragTrees	IML	[16]
BRL	IML	[18]
Anchors	Local	[22]
LORE	Local	[21]

The nine data sets selected for this study have been carefully selected to represent a mix of binary and multi-class problems, to exhibit different levels of class imbalance (no artificial balancing will be applied), to be a mixture of discrete and continuous features, and to be a contextual fit for XAI (i.e., credit and personal data) where possible. The data sets are detailed in Table 3. All are publicly available and taken from the UCI Machine Learning Repository [36] except lending (Kaggle) and rcdv (ICPSR; https://tinyurl.com/y8qvcgwu (30 October 2019)). Three of these data sets (adult, lending and rcdv) are those used in [22] and, therefore, align with precedents and provide direct comparisons to state-of-the-art methods Anchors and LIME. The exceptionally large lending data set has been subsampled. The number of instances in the original data set is 842,000 and was downsampled to $N = 2105$ for these experiments. Training and test data sets were sampled without replacement into partitions of size 70 and 30% of the original data set.

Table 3. Data sets used in the experiments.

Data Set	Target	Classes	Class Balance	Features	Categorical	N
adult	income	2	0.77:0.23	14	9	48,842
bank	y	2	0.89:0.11	20	11	45,307
car	acceptability	2	0.71:0.29	7	7	1728
cardio	NSP	3	0.78:0.14:0.08	22	1	2126
credit	A16	2	0.57:0.43	16	10	690
german	rating	2	0.70:0.30	21	14	1000
lending	loan_status	2	0.79:0.21	75	9	2105
nursery	decision	4	0.33:0.33:0.31:0.02	9	9	12,958
rcdv	recid	2	0.38:0.62	19	19	18,876

3.3. Quantitative Study

There is a very strong case that coverage and precision are not appropriate quality metrics for explanations-based research [31,32]. Coverage is trivially maximised by critically under-fitting solutions. For example, the *null* rule $\mathcal{X} \implies g(\mathbf{x})$ (all inputs result in the given output) is critically under-fitting, yet scores 1.0 for coverage. Precision, conversely, is trivially maximised by critically over-fitting solutions. For example, the "tautological" rule $\{feature_1 = x_1, \ldots, feature_P = x_P\} \implies g(\mathbf{x})$ (the unique attributes of the explanandum result in the given output) is critically over-fitting yet scores 1.0 for precision.

These metrics are absolutely ubiquitous throughout the ML, statistical and data mining literature, which might explain their continued application in XAI experimental research. This research prefers *reliability* and *exclusive coverage*, first proposed in [31], because they penalise any explanations that approach the ill-fitting situations described above. However, to assist the user in understanding the utility of these novel metrics, both sets of results (novel and traditional) are presented. The rule length (cardinality of the rule antecedent) is also measured. These metrics are supplemented by the reliability floor and the rule length floor. The reliability floor is the proportion of evaluated explanations that clear the threshold of 0.75 reliability, and the rule length floor is the proportion of explanations with a length greater than zero. Both of these supplementary statistics are useful for quantifying the prevalence of over- and under-fitting. These pathological behaviours can easily be masked when only looking at aggregate scores (means, mean ranks, etc.). We will also present the fidelity scores that reveal when methods/proxy models do not agree with the black box reference model.

The computational complexity will be compared using the mean time (sec) to generate a single explanation. The authors of [19,22] state that their methods take a few seconds to a few minutes to generate an explanation. We conjecture that something less than thirty seconds would be considered acceptable for many Human–in–the–Loop processes because each explanation requires further consideration prior to completion of a downstream task. Consideration and completion steps would likely be much longer than this simple time threshold.

Significance (where required) shall be evaluated with the modified Friedman test, given in [40]. The Friedman test [41] is a non-parametric equivalent to ANOVA and an extension of the rank sum test for multiple comparisons. The null hypothesis of this test is that the mean ranks for all groups are approximately equal. In these experiments, the groups are the competing algorithms. The alternative hypothesis is that at least two mean ranks are different.

On finding a significant result, the pairwise, post-hoc test can be used to determine which of the methods perform significantly better or worse than the others. It is sufficient for this study to demonstrate whether the top scoring method was significantly greater than the second place method. Note, however, that the critical value is applied as if all the pairwise comparisons were made. The critical value for a two-tailed test with the

Bonferroni correction for six groups is $\frac{0.025}{6} = 0.0042$. The winning algorithm is formatted in boldface only if the results are significant.

4. Discussion

This section presents the main results of the experimental research. Supplementary results are available from our github repository https://tinyurl.com/yxuhfh4e (5 March 2021). Note, all the Friedman tests yielded significant results. Consequently, these results are omitted and we proceed directly to the pairwise, post-hoc tests between the top two methods. These post-hoc tests will help to determine if there is an overall leading method.

4.1. Fidelity

Fidelity (the agreement rate between the explanations' consequent and the reference model's output), is given in Table 4. Only gtb-HIPS and Anchors are guaranteed to be locally accurate by means of their algorithmic steps. Unfortunately, it was not possible to collect the fidelity scores for the LORE method. The computation time of the (very long-running) method, which makes it prohibitive to re-run the experiments. However, the fidelity of LORE is listed in the originating paper [21] as 0.959 ± 0.17 for the adult data set, 0.988 ± 0.07 for the german data set, and 0.992 ± 0.03 for a third data set not used in this investigation. LORE is assumed to reach this level of fidelity for other data sets used in these experiments.

Table 4. Fidelity of Bayesian rule list (BRL), defragTrees and inTrees to the black box (gradient boosted tree (GBT)) model.

Data	gbt-HIPS	Anchors	BRL	defragTrees	inTrees
adult	1.00	1.00	0.92	0.93	0.89
bank	1.00	1.00	0.95	0.92	0.94
car	1.00	1.00	0.96	0.75	0.96
cardio	1.00	1.00	0.89	0.85	0.91
credit	1.00	1.00	0.91	0.89	0.94
german	1.00	1.00	0.76	0.71	0.81
lending	1.00	1.00	N/A	0.94	NA
nursery	1.00	1.00	0.99	0.82	0.68
rcdv	1.00	1.00	0.85	0.83	0.86

It must be noted that poor fidelity with the black box model is a critical flaw. An explanation in the form of a classification rule is not fit for purpose if the consequent does not match the target class. On this point, there can be little debate because a key requirement, local accuracy, is violated. On the other hand, the tolerance for anything less than perfect fidelity is a domain-specific question. This tolerance will depend on the cost or inconvenience of failing to explain any given instance. So, we make only the following assertion as to what is an acceptable score: it would be surprising to find levels as low as 0.90 permissible in critical applications. At this level, one in ten explanations is unusable.

4.2. Generalisation

Good performance on the exclusive coverage metric indicates that the explanations generalise well to new data. Such rules cover a large proportion of data from the target distribution without covering large numbers of instances that the black box classified differently than the explanandum instance.

A cursory visual inspection the coverage (Figure 3) does not reveal any obvious pattern. While there is no overall winning algorithm for coverage, BRL and inTrees are each strongly in the lead for three out of the nine data sets, and gbt-HIPS for two of the remaining three data sets. On the other hand, visual analysis of the exclusive coverage score distribution (Figure 4) shows that gbt-HIPS is often leading or a close runner-up.

The lead that BRL had for simple coverage is completely forfeit. In fact BRL has the lowest exclusive coverage for five out of nine data sets. Furthermore, the inTrees method no longer has the lead in any data set, except for nursery under the exclusive coverage measure. The tabulated mean and mean ranks of these data (in the supplementary materials) support this visual analysis and show that gbt-HIPS takes the lead for six out of the nine data sets. This result suggests that BRL and inTrees generate explanations that are too general, while gbt-HIPS explanations are robust. This diagnosis is borne out by results from the rule length floor statistic (to follow).

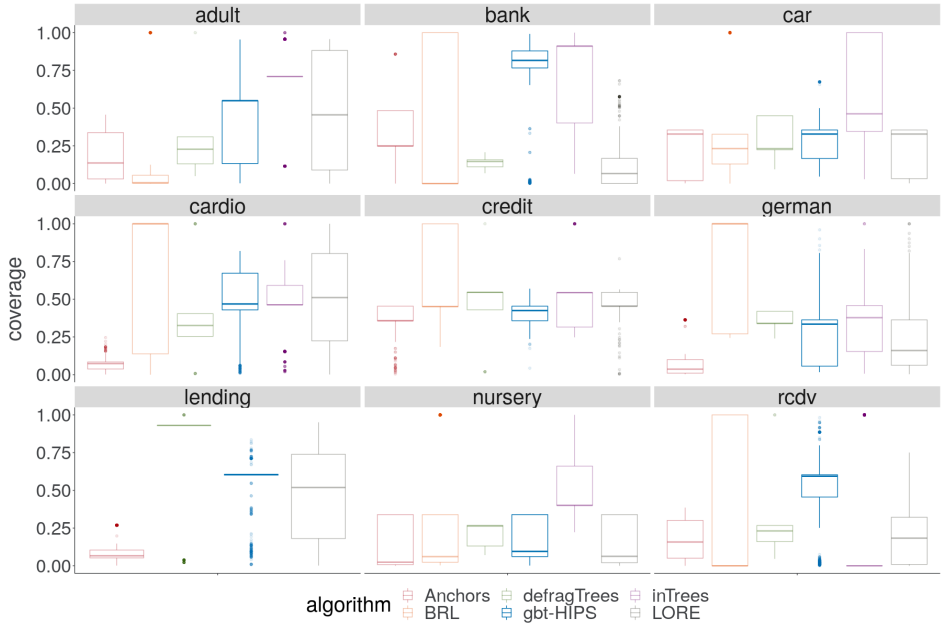

Figure 3. Distribution of coverage for stochastic GBT model explanations.

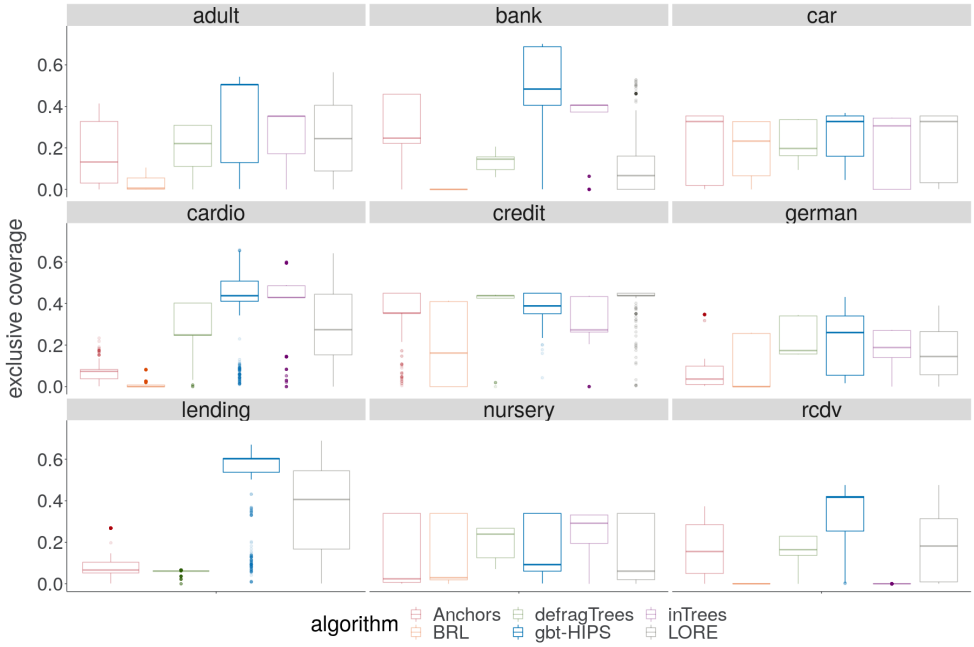

Figure 4. Distribution of exclusive coverage for stochastic GBT model explanations.

Significance tests between the top two ranking methods are shown in Table 5. gbt-HIPS ranked first on five data sets, joint first (no significant difference between first and second) on the car data set, second on the german data set, and third out of six methods on the remaining two data sets, making gbt-HIPS the very clear lead.

Table 5. Exclusive coverage significance tests: top two methods by mean rank (mrnk) for the stochastic GBT model.

Data	1st	Mrnk	2nd	Mrnk	N	z	p Value
adult	**gbt-HIPS**	1.95	inTrees	2.34	1000	18.1	≈0
bank	**gbt-HIPS**	1.54	inTrees	2.24	1000	30.1	≈0
car	defragTrees	3.00	gbt-HIPS	3.17	517	1.4	0.0781
cardio	**gbt-HIPS**	1.92	inTrees	2.21	637	3.2	<0.00417
credit	LORE	2.20	defragTrees	2.59	206	2.1	0.0169
german	**defragTrees**	2.31	gbt-HIPS	2.80	299	3.2	<0.00417
lending	**gbt-HIPS**	1.23	LORE	2.01	631	10.1	≈0
nursery	**inTrees**	2.69	defragTrees	3.17	1000	13.4	≈0
rcdv	**gbt-HIPS**	1.50	defragTrees	2.82	1000	37.1	≈0

4.3. Reliability

Good performance on the reliability metric indicates that, for the target distribution, a high proportion of instances covered by the explanation will receive the same classification from the black box model as was given to the explanandum. At the same time, the end user can be certain that the rule does not cover a trivially small region of the input space.

A cursory visual inspection of the precision (Figure 5) demonstrates the trade-off between precision and coverage. The BRL, inTrees and defragTrees methods that had scored relatively well for coverage do not deliver state-of-the-art precision on any data set. Both precision and reliability (Figure 6) score distributions show that Anchors and gbt-HIPS vie for first position over almost all of the data sets. Anchors appears to have a slight advantage for precision while gbt-HIPS appears to do better for reliability. The placement is often so close that it requires recourse to the tabulated results (supplementary materials) and the significance tests to be certain of the leading method.

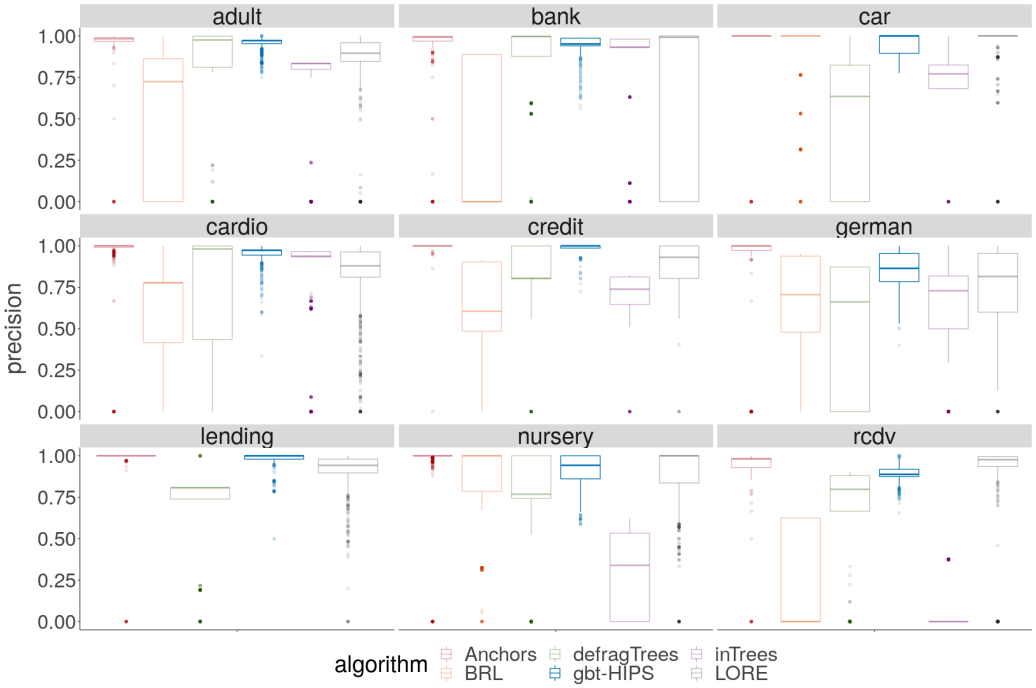

Figure 5. Distribution of precision for stochastic GBT model explanations.

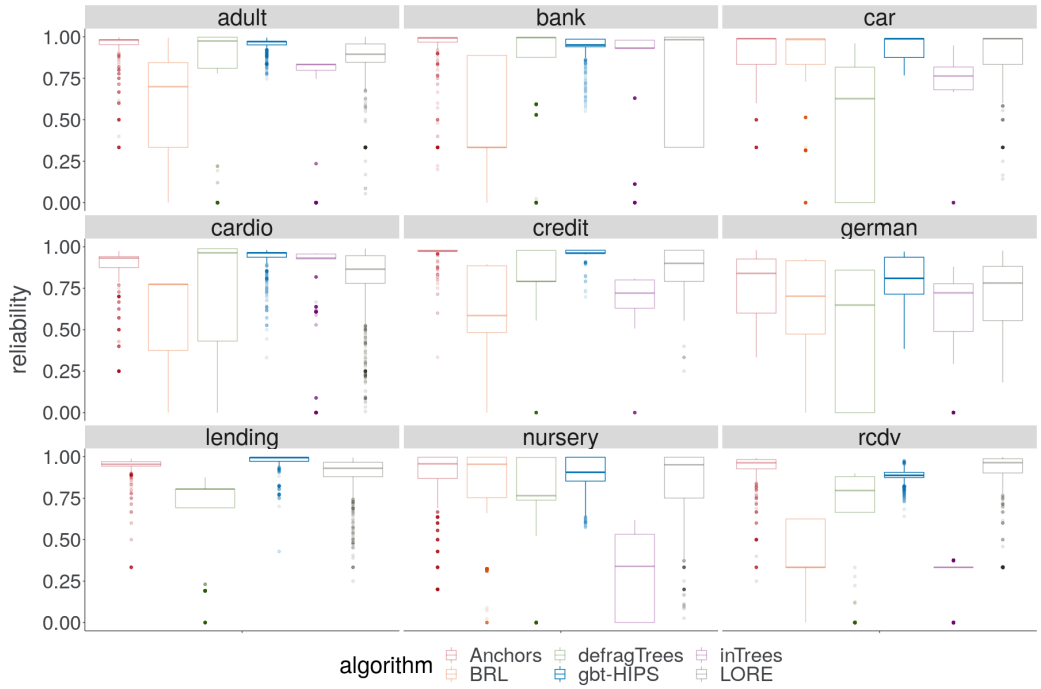

Figure 6. Distribution of reliability for stochastic GBT model explanations.

The results of hypothesis tests of the pairwise comparisons for the top two ranking methods are shown in Table 6. The tests seem to show that Anchors is leading for reliability on three out of the nine data sets, joint first (no significant difference between first and second place) on a further two data sets, and second on a further two data sets. gbt-HIPS appears to be the second place method, leading on two data sets, and joint first on a further three data sets. These results, it seems, are inconsistent with the tabulated and visualised mean scores for reliability.

Table 6. Reliability significance tests: top two methods by mean rank (mrnk) for the stochastic GBT model.

Data	1st	Mrnk	2nd	Mrnk	N	*z*	*p* Value
adult	**Anchors**	2.19	defragTrees	2.62	1000	19.4	≈0
bank	**Anchors**	2.23	LORE	2.83	1000	26.3	≈0
car	Anchors	2.60	gbt-HIPS	2.61	517	0.1	0.4266
cardio	**gbt-HIPS**	1.95	defragTrees	2.58	637	7.04	<0.00417
credit	Anchors	1.69	gbt-HIPS	1.94	206	1.4	0.00865
german	gbt-HIPS	2.18	Anchors	2.56	299	2.5	0.00671
lending	**gbt-HIPS**	1.23	Anchors	2.18	631	13.1	≈0
nursery	**BRL**	2.28	Anchors	2.61	1000	9.21	≈0
rcdv	**Anchors**	1.74	LORE	1.89	1000	4.3	<0.00417

These inconsistencies are, unfortunately, an artefact of the choice of significance test, which is non-parametric and, therefore, insensitive to outliers. Specifically, the long tail of under-fitting instances visible as colour saturated dots in the lower parts of each facet of Figure 6. For gbt-HIPS, instead almost the entire set of scores occupies a narrow band near to the upper bound for reliability. It is for this reason that the reliability floor statistic (Figure 7) is so enlightening. The reliability floor quantifies the propensity to over-fit by measuring the proportion of explanations in the test set that scored above the threshold. Over-fitting explanations are too granular and cover too few instances. Furthermore, a significant number of explanations score zero, demonstrating a critical over-fit. That is, an explanation that covers only the explanandum but not a single instance in the held out

set. gbt-HIPS leads on all nine data sets for reliability floor. The reliability floor scores are presented visually in Figure 7 and tabulated in the supplementary materials.

Figure 7. 0.75 reliability floor for stochastic GBT model explanations.

4.4. Interpretability

While antecedent length is not an absolute measure, it can be used to compare the relative understandability of CR-based explanations. Significance tests do not form a part of this analysis for the following reason. Even though short rules are the most desirable, a score of zero length (the null rule) is highly undesirable and a sign of under-fitting. The significance test, based on mean ranks of rule lengths (in ascending order) will reward methods with this pathological condition. So, rather than fabricating a new mode of testing, this research relies on the evidence of the visual analysis, and the rule length floor statistic. Anchors and gbt-HIPS are guaranteed never to return a zero length rule via their algorithmic steps. All of the globally interpretable methods, on the other hand, can return the zero length null rule if none of the rules in their list are found to cover the explanandum. It would be highly unexpected for LORE to return a null rule but there is no formal guarantee of this behaviour and, very occasionally, it does occur.

On reviewing the rule length results visually in Figure 8, it is encouraging to note that gbt-HIPS never generates either the longest rules or suspiciously short rules. Interestingly Anchors, LORE and gbt-HIPS track one another (very approximately) for mean rule length (supplementary materials) over all the data sets, which might suggest some level of commonality in their outputs. The BRL method, on the other hand, generates the longest rules for four out of five data sets. The defragTrees method generates the longest rules on a further two. In these cases, the rule lengths measured suggest that a large number of instances are explained by rules that are some way down the CRL, resulting in concatenation. The BRL method also generates the shortest explanations for the credit (0.38 ± 0.05), and german (1.23 ± 0.05) data sets. The inTrees method generates the shortest explanation for the adult (0.34 ± 0.04) data set, and the bank (1.12 ± 0.03) data set. Values less than 1.00 indicate a critical tendency to under-fit, with a high prevalence of zero length rules that have deflated the mean length to the point of no longer being a meaningful measure. This behaviour is revealed and quantified by the rule length floor results (Table 7). The rule length floor statistic with a threshold of 0 is simply the fraction of explanations that have a length greater than 0. These results explain the very large contrast between traditional coverage and exclusive coverage for these methods and data sets. This statistic also makes clear the utility of using exclusive coverage for evaluating experiments in the XAI setting.

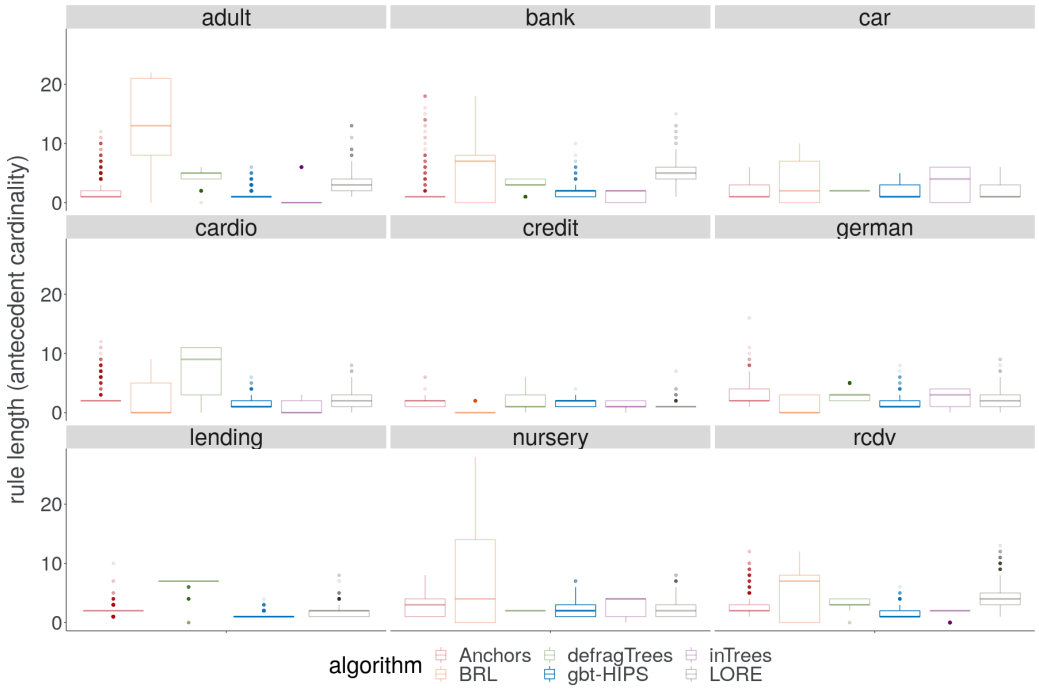

Figure 8. Distribution of rule cardinality for stochastic GBT model explanations.

Table 7. Rule length floor (threshold 0) for the stochastic GBT model. This statistic shows the proportion of explanations with antecedent length > 0.

Data	gbt-HIPS	Anchors	BRL	defragTrees	inTrees	LORE
adult	1.00	1.00	0.96	1.00	0.06	1.00
bank	1.00	1.00	0.56	1.00	0.56	1.00
car	1.00	1.00	0.57	1.00	0.75	1.00
cardio	1.00	1.00	0.29	0.99	0.47	0.99
credit	1.00	1.00	0.19	1.00	0.93	0.99
german	1.00	1.00	0.46	1.00	0.93	0.99
lending	1.00	1.00	N/A	0.99	N/A	1.00
nursery	1.00	1.00	0.51	1.00	0.79	1.00
rcdv	1.00	1.00	0.65	1.00	0.73	1.00

4.5. Computation Time

For this part of the results analysis, the statistic of interest is simply the arithmetic mean computation time for all the explanations. The mean computation time is presented in Table 8. There are no significance tests since it is sufficient to show that the mean time per explanation is thirty seconds or less (shorter than the time prescribed by [22]). For gbt-HIPS, the range of mean times per explanation was

- longest—25.73 ± 0.35 (s) for the *adult* data set;
- shortest—1.30 ± 0.02 (s) for the *car* data set.

Based upon this simple, threshold-based assessment, while gbt-HIPS is not the fastest method in this study, the threshold is met for all data sets. BRL, defragTrees and inTrees are fast or very fast for all data sets since once the model is built, classification and explanation are a result of the same action. However, it must be noted that these methods have not performed well on the main metrics of interest. LORE is universally the longest running method, as a result of a genetic algorithmic step that results in thousands of calls to the target black box model. The run-times were, unfortunately, too long to be considered useful in a real-world setting.

Table 8. Mean elapsed time of explanations of the stochastic GBT model.

Data	gbt-HIPS	Anchors	BRL	defragTrees	inTrees	LORE
adult	25.73 ± 0.35	0.24 ± 0.01	**0.01** ± 0.00	6.62 ± 0.00	0.05 ± 0.00	585.87 ± 0.72
bank	21.67 ± 0.81	0.18 ± 0.01	3.83 ± 0.00	5.32 ± 0.00	**0.02** ± 0.00	506.95 ± 0.52
car	1.51 ± 0.03	0.07 ± 0.00	**0.01** ± 0.00	0.05 ± 0.00	0.06 ± 0.00	12.57 ± 0.02
cardio	3.28 ± 0.05	0.65 ± 0.02	**0.02** ± 0.00	0.09 ± 0.00	0.07 ± 0.00	19.16 ± 0.06
credit	1.30 ± 0.02	0.28 ± 0.01	0.09 ± 0.00	**0.02** ± 0.00	0.03 ± 0.00	6.64 ± 0.01
german	2.61 ± 0.08	0.48 ± 0.01	0.19 ± 0.00	**0.04** ± 0.00	0.08 ± 0.00	11.31 ± 0.02
lending	2.58 ± 0.05	1.67 ± 0.02	N/A	**0.04** ± 0.00	N/A	27.37 ± 0.04
nursery	3.24 ± 0.06	0.24 ± 0.01	**0.00** ± 0.00	0.01 ± 0.00	0.01 ± 0.00	24.14 ± 0.08
rcdv	8.38 ± 0.13	0.26 ± 0.00	**0.01** ± 0.00	2.25 ± 0.00	**0.01** ± 0.00	114.34 ± 0.14

5. Conclusions and Future Work

In this paper we presented gbt-HIPS, a novel, greedy, heuristic method for explaining gradient boosted tree models. To the best of our knowledge, these models have not previously been the target of a model-specific explanation system. Such explanation systems are quite mature for neural networks, including deep learning methods, but only recently have ensembles of decision trees been subject to similar treatment. We conjecture that the non-differentiable, non-parametric nature of decision trees is the cause of this gap. Our method not only provides a statistically motivated approach to decision path and node activation but also produces explanations that more closely adhere to generally accepted ideals of explanation formats than any previous work. In addition, we have presented an experimental framework that helps to quantify specialised under- and over-fitting problems that can occur in the XAI setting.

As a future direction for research, we suggest a focus on multi-objective optimisation and global search methods such as genetic algorithms to replace the simple, greedy, heuristic rule-merge step. Such a procedure would benefit the method by generating a non-dominated Pareto set of explanations that captures the breadth of optimisation targets—reliability, generality, rule length and accumulated path weight.

Supplementary Materials: The following are available at www.mdpi.com/xxx/s1.

Author Contributions: Conceptualisation, J.H. and M.M.G.; methodology, J.H., M.M.G. and R.M.A.A.; software, J.H.; validation, J.H.; investigation, J.H.; data curation, J.H.; writing—original draft preparation, J.H.; writing—review and editing, M.M.G. and R.M.A.A.; supervision, M.M.G. and R.M.A.A.; project administration, J.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are openly available in our github repostory https://tinyurl.com/yxuhfh4e (5 March 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Friedman, J. Greedy Function Approximation: A Gradient Boosting Machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
2. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
3. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
4. Dorogush, A.V.; Ershov, V.; Gulin, A. CatBoost: Gradient boosting with categorical features support. In Proceedings of the Thirty-First Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
5. Jovanovic, M.; Radovanovic, S.; Vukicevic, M.; Van Poucke, S.; Delibasic, B. Building interpretable predictive models for pediatric hospital readmission using Tree-Lasso logistic regression. *Artif. Intell. Med.* **2016**, *72*, 12–21. [CrossRef] [PubMed]

6. Turgeman, L.; May, J.H. A mixed-ensemble model for hospital readmission. *Artif. Intell. Med.* **2016**, *72*, 72–82. [CrossRef]
7. Letham, B.; Rudin, C.; McCormick, T.H.; Madigan, D. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *Ann. Appl. Stat.* **2015**, *9*, 1350–1371. [CrossRef]
8. Chajewska, U. Defining Explanation in Probabilistic Systems. *arXiv* **2013**, arXiv:1302.1526.
9. Binns, R.; Van Kleek, M.; Veale, M.; Lyngs, U.; Zhao, J.; Shadbolt, N. 'It's Reducing a Human Being to a Percentage': Perceptions of Justice in Algorithmic Decisions. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, Montreal, QC, Canada, 21–26 April 2018; pp. 1–14.
10. Larson, J.; Mattu, S.; Kirchner, L.; Angwin, J. *How We Analyzed the COMPAS Recidivism Algorithm*; Technical Report; ProPublica: New York, NY, USA, 2016.
11. Dickerson, S.; Haggerty, P.; Hall, P.; Cox, B.; Kannan, A.R.; Kulkarni, R.; Prochaska, K.; Schmidt, N.; Wiwczaroski, M. *Machine Learning-Considerations for Fairly and Transparently Expanding Access to Credit;* H2O.ai, Inc.: Mountain View, CA, USA, 2020.
12. Press, G. *X Equifax and SAS Leverage AI and Deep Learning to Improve Consumer Access to Credit*; Forbes: Jersey City, NJ, USA, 2017.
13. Mathew, A. Credit Scoring Using Logistic Regression. Master's Thesis, San Jose State University, San Jose, CA, USA, 2017.
14. Gunning, D. *Explainable Artificial Intelligence (XAI)*; Defense Advanced Research Projects Agency (DARPA): Arlington, VA, USA, 2017.
15. Pasquale, F. *The Black Box Society: The Secret Algorithms that Control Money and Information*; Harvard University Press: Cambridge, MA, USA, 2015.
16. Hara, S.; Hayashi, K. Making Tree Ensembles Interpretable: A Bayesian Model Selection Approach. *arXiv* **2016**, arXiv:1606.09066.
17. Deng, H. Interpreting tree ensembles with intrees. *Int. J. Data Sci. Anal.* **2014**, *7*, 277–287. [CrossRef]
18. Letham, B. Statistical Learning for Decision Making: Interpretability, Uncertainty, and Inference. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2015.
19. 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, San Francisco, CA, USA, 13–17 August 2016; pp. 1135–1144.
20. Lundberg, S.M.; Lee, S.I. A Unified Approach to Interpreting Model Predictions. *arXiv* **2017**, arXiv:1705.07874.
21. Guidotti, R.; Monreale, A.; Ruggieri, S.; Pedreschi, D.; Turini, F.; Giannotti, F. Local Rule-Based Explanations of Black Box Decision Systems. *arXiv* **2018**, arXiv:1805.10820.
22. Ribeiro, M.T.; Singh, S.; Guestrin, C. Anchors: High-Precision Model-Agnostic Explanations. *AAAI* **2018**, *18*, 1527–1535.
23. Zhu, J.; Liapis, A.; Risi, S.; Bidarra, R.; Youngblood, G.M. Explainable AI for Designers: A Human-Centered Perspective on Mixed-Initiative Co-Creation. In Proceedings of the 2018 IEEE Conference on Computational Intelligence and Games (CIG), Maastricht, The Netherlands, 14–17 August 2018.
24. Rudin, C. Please Stop Explaining Black Box Models for High Stakes Decisions. *arXiv* **2018**, arXiv:1811.10154.
25. Gosiewska, A.; Biecek, P. Do Not Trust Additive Explanations. *arXiv* **2020**, arXiv:1903.11420.
26. Fen, H.; Song, K.; Udell, M.; Sun, Y.; Zhang, Y. Why should you trust my interpretation? Understanding uncertainty in LIME predictions. *arXiv* **2019**, arXiv:1904.12991.
27. Slack, D.; Hilgard, S.; Jia, E.; Singh, S.; Lakkaraju, H. Fooling LIME and SHAP: Adversarial Attacks on Post hoc Explanation Methods. *arXiv* **2020**, arXiv:1911.02508.
28. Islam, S.R.; Eberle, W.; Ghafoor, S.K. Towards Quantification of Explainability in Explainable Artificial Intelligence Methods. In Proceedings of the Thirty-Third International FLAIRS Conference, North Miami Beach, FL, USA, 17–20 May 2020.
29. Molnar, C. *Interpretable Machine Learning*; Lulu Press: Morrisville, NC, USA, 2019.
30. Wachter, S.; Mittelstadt, B.; Russell, C. Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. *Harv. J. Law Technol.* **2017**, *31*, 841. [CrossRef]
31. Hatwell, J.; Gaber, M.M.; Azad, R.M.A. CHIRPS: Explaining random forest classification. *Artif. Intell. Rev.* **2020**, *53*, 5747–5788. [CrossRef]
32. Hatwell, J.; Gaber, M.M.; Azad, R.M.A. Ada-WHIPS: Explaining AdaBoost classification with applications in the health sciences. *BMC Med. Inform. Decis. Mak.* **2020**, *20*, 1–25. [CrossRef] [PubMed]
33. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
34. Freund, Y.; Schapire, R.E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
35. Miller, T. Explanation in Artificial Intelligence: Insights from the Social Sciences. *arXiv* **2017**, arXiv:1706.07269.
36. Dheeru, D.; Karra Taniskidou, E. *UCI Machine Learning Repository*; School of Information and Computer Sciences, University of California: Irvine, CA, USA, 2017.
37. Doshi-Velez, F.; Kim, B. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv* **2017**, arXiv:1702.08608.
38. Wang, T.; Rudin, C.; Doshi-Velez, F.; Liu, Y.; Klampfl, E.; MacNeille, P. A Bayesian Framework for Learning Rule Sets for Interpretable Classification. *J. Mach. Learn. Res.* **2017**, *18*, 37.
39. Lakkaraju, H.; Bach, S.H.; Leskovec, J. Interpretable Decision Sets: A Joint Framework for Description and Prediction. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1675–1684.
40. Demšar, J. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* **2006**, *7*, 1–30.
41. Hutchinson, T. On the generalised friedman test. *Comput. Stat. Data Anal.* **1996**, *21*, 473–476. [CrossRef]