

# VisRuler: Visual Analytics for Extracting Decision Rules from Bagged and Boosted Decision Trees

A. Chatzimparmpas<sup>1</sup> , R. M. Martins<sup>1</sup> , and A. Kerren<sup>1,2</sup>

<sup>1</sup>Department of Computer Science and Media Technology, Linnaeus University, Sweden

<sup>2</sup>Department of Science and Technology, Linköping University, Sweden

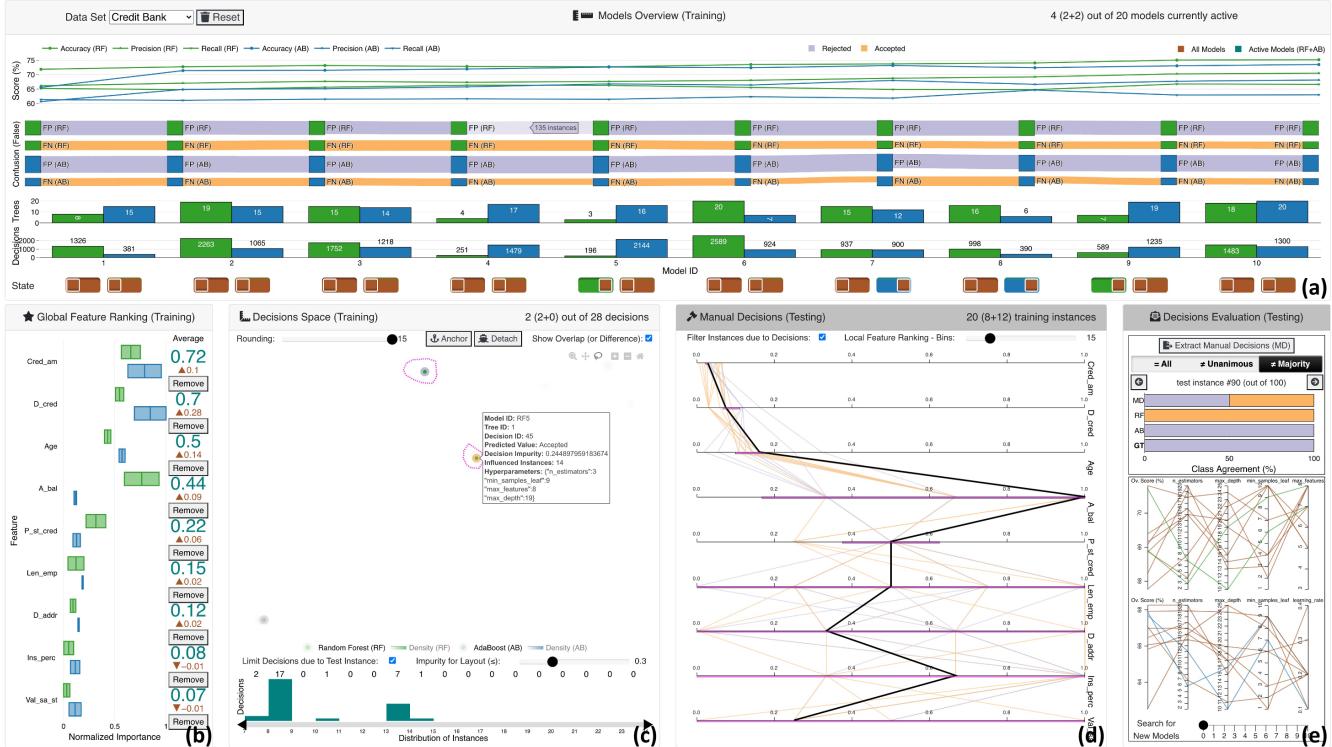


Figure 1: Extracting decision rules for manual evaluation with VISRULER: (a) the panel with various visual metaphors for selecting performant and diverse models; (b) the box plot for feature selection according to per algorithmic importance; (c) the visual embedding of computed decisions that training instances fall in due to their values; (d) the vertical parallel coordinates plot that summarizes the rules with value ranges for each feature and highlights the current test instance; and (e) the horizontal stacked bar chart for revealing the class agreement of each model against the manual decisions, together with the parallel coordinates plots for tuning hyperparameters and training new models.

## Abstract

Bagging and boosting are two popular ensemble methods in machine learning (ML) that produce many individual decision trees. Due to the inherent ensemble characteristic of these methods, they typically outperform single decision trees or other ML models in predictive performance. However, numerous decision paths are generated for each decision tree, increasing the overall complexity of the model and hindering its use in domains that require trustworthy and explainable decisions, such as finance, social care, and health care. Thus, the interpretability of bagging and boosting algorithms—such as random forests and adaptive boosting—reduces as the number of decisions rises. In this paper, we propose a visual analytics tool that aims to assist users in extracting decisions from such ML models via a thorough visual inspection workflow that includes selecting a set of robust and diverse models (originating from different ensemble learning algorithms), choosing important features according to their global contribution, and deciding which decisions are essential for global explanation (or locally, for specific cases). The outcome is a final decision based on the class agreement of several models and the explored manual decisions exported by users. Finally, we evaluate the applicability and effectiveness of VisRuler via a use case, a usage scenario, and a user study.

## CCS Concepts

- Human-centered computing → Visualization; Visual analytics;
- Machine learning → Supervised learning;

## 1. Introduction

Ensemble learning (EL) [Zho09] is a well-established area of machine learning (ML) that strives for better performance by merging the predictions from various ML models. Three prominent methods for building ensembles are [SR18]: bagging [Bre96], boosting [FS96, Sch90], and stacking [Wol92]. Bagging requires training many decision trees on separate groups of instances of a data set and taking the average of their predictions [Bre96]. Boosting includes attaching weak classifiers (e.g., decision stumps or shallow decision trees) sequentially, each improving the predictions made by the previous models [FS96, Sch90]. Stacking involves fitting many base models from different algorithms on the same data set and using a metamodel to combine their results [Wol92]. The common ground between bagging and boosting methods is that they incorporate ML algorithms that produce numerous decision trees [KS08], such as random forests (RF) [Bre01a] and adaptive boosting (AB) [FSA99], respectively. The decision paths stemming from bagged or boosted decision trees are the target of the visual analytics (VA) approach proposed in this paper.

The popularity of RF and AB is confirmed by their success in solving typical supervised classification problems, which constitute the majority of problems in the real world [OM99, WOBM17]. An in-depth study [FDCBA14] that estimates the performances of 179 algorithms of various types [DG17] concludes that bagged decision trees of RF are better than other (types of) algorithms, such as deep learning approaches. Despite their remarkable predictive power, a crucial concern for algorithms that generate many decision trees is *interpretability*. Breiman [Bre01b], for instance, indicates that RF models, while superb predictors, receive a low rating regarding their interpretability. As ML models can provide incorrect predictions [CLG\*15], ML experts have to check whether the model functions properly [TKC17]. Also, domain experts in critical fields need to understand how a specific prediction has been reached in order to trust in ML [ZC18]. For example, in medicine, a physician might not rely on a model without explanations of how and why it forms a prediction, since patient lives are at risk [RSG16, HTF01, LBL16]. Or, in the financial domain, declined decisions for loan applicants require additional transparency with the precise justification of the outcome [SYX\*20]. Thus, one research question that remains open is: **(RQ1)** How do bagged decision trees' learned rules differ from boosted decision trees, and is there any potential benefit in combining them, regarding interpretability and predictive performance?

The interpretation of ML models typically happens either at a global or a local level [KCKS19]. Global approaches intend to explain the ML model as a whole [Lip18], assisting domain experts in exploring the general impact of each decision and gaining confidence in the produced predictions. On the other hand, local approaches aim to provide case-based reasoning [DLH19, CPC19], allowing domain experts to review a prediction and trace its decision path in order to conclude if the decision rule, and consequently the prediction, is trustworthy [Wei19]. Nevertheless, comparing numerous alternative decision paths without the support of an intelligent system is a time-consuming and resource-heavy procedure. For example, to scan the list of test instances rapidly and investigate specific instances of interest from multiple perspectives (e.g., outliers and borderline cases) can be crucial [KRS14]. One

research question that arises from these explanations—inspired by Streeb et al. [SMS\*21]—is: **(RQ2)** How can visualizations and VA tools/systems facilitate the externalization of domain knowledge?

In this paper, we present VISRULER (see Figure 1), a VA tool that addresses the research questions described above by supporting the exploratory combination of decisions from two closely-related ML algorithms (i.e., RF and AB). VISRULER uses validation metrics for picking performant and diverse models and combines the decision paths from bagged and boosted trees to extract insightful and interpretable rules. Our contributions consist of the following:

- a visual analytic workflow for defining a methodical way of evaluating decisions (cf. Figure 2 described in Section 4);
- a prototype VA tool, called VISRULER, that applies the suggested workflow with coordinated views that support the joint effort between ML experts and domain experts for extracting rules and making decisions, respectively;
- a use case and a usage scenario, applying real-world data, that validate the effectiveness of utilizing both bagged and boosted decision trees at the same time; and
- a user study that showed promising results.

## 2. Related Work

According to a recent survey [SMS\*21] that has extensively analyzed tree- and rule-based classification, several VA systems have been developed for this topic in the InfoVis and VA communities. However, most of these tools do not employ algorithms and measures (except for the accuracy metric) in order to compare model quality [SMS\*21]. This section reviews prior work on the interpretation of bagged and boosted decision trees and the more general tools for tree- and rule-based visualization, comparing them with VISRULER to highlight our tool's novelty.

**Interpretation of Bagged Decision Trees.** As in VISRULER, relevant works that utilize bagging methods use the RF algorithm to produce decision trees [NP21a, NWWH19, ZWLC19, NP21b]. iForest [ZWLC19] provides users with tree-related information and an overview of the involved decision paths for case-based reasoning, with the goal of revealing the model's working internals. However, iForest can be used only for binary classification, while VISRULER can be used with multi-class data sets (as in the use case of Section 4). Also, the feature flow, a node-link diagram, suffers from scalability issues (a challenge only partially overcome with aggregation). Our tool's approach with dimension reduction employed for clustering all decisions extracted by multiple models enables users to gain insights into the relation of large quantities of rules. Therefore, VISRULER allows users to mine rules for both a particular class outcome and in connection to a specific case. ExMatrix [NP21a] is another VA tool for RF interpretation that operates using a matrix-like visual representation, facilitating the analysis of a model and connecting rules to classification results. While the scalability is good, it does not cover the task of finding similarities between decisions from diverse models and algorithms. In conclusion, none of the above works have experimented with the fusion of bagged and boosted decision trees, and in particular, with visualizing both tree types in a joint decision space to observe their dissimilarity, which can result in unique and undiscovered decisions.

**Interpretation of Boosted Decision Trees.** Special attention has been given to boosted decision trees with VA tools for diagnosing the training process of boosting methods [LXL<sup>\*</sup>18, HLLW19, WZWY21] and interpreting their decisions [XCC<sup>\*</sup>21]. Closer to our work, GBMVVis [XCC<sup>\*</sup>21] aims to reveal the structure and properties of Gradient boosting [Fri01], enabling users to examine the importance of features and follow the data flow for different decisions. A node-link diagram may limit its scalability to monitor hundreds or thousands of decisions concurrently, as opposed to VISRULER. Furthermore, our novel parallel coordinates plot adaptation allows users to instantly combine rules and observe their differences to identify unique decisions. BOOSTVis [LXL<sup>\*</sup>18] employs views such as a temporal confusion matrix visualization for verifying the performance changes of the model, a t-SNE [vdMH08] projection for inspecting the instances, and a node-link diagram for examining the rules. Through GBRTVis [HLLW19], users can explore Gradient boosting [Fri01] with a node-link diagram for the rules, the instances distribution shown in a treemap, and continuously monitoring the loss function. VISTB [WZWY21] contains a redesigned temporal confusion matrix to track the per-instance prediction during the training process. It also enables the comparison of the impact of individual features over iterations. These VA systems focus on the online training of boosting methods and aim to assist in feature selection and hyperparameter tuning. While these problems are (partially) tackled by our tool, we concentrate on interpreting the decisions from bagged and boosted decision trees and comparing them across models.

**Tree- and Rule-based Model Visualization.** Existing work on single decision tree visualization has experimented with different visualization techniques, such as node-link diagrams [vdEvW11, NHS00, LJC16, BN01, PNWG17, BvLH<sup>\*</sup>11, SCS04, MGT<sup>\*</sup>03, BKSS14, C19], treemaps [MLMP18, GGPPS13], icicle plots [PSMD14, AEK00], star coordinates [TM03b, TM03a], and 2D scatter-plot matrices [Do07]. These techniques do not generalize well when exploring multiple decision trees, which is VISRULER’s primary design goal. Visualizing the surrogate models to approximate the behaviors of the original models, either globally or locally, is another branch of related works [CB20, DCB19, HC00, DB21, WFH<sup>\*</sup>01, YNB21, EAM14]. Rule-based visualizations have also been deployed for the interpretation of complex neural networks [MJEP<sup>\*</sup>21, MQB19, JLL<sup>\*</sup>20]. Nevertheless, these models differ due to the lack of inherent decisions that could be extracted directly from the bagged and boosted decision trees. The core mechanism of bagging and boosting methods is the generation of decisions based on the training data, which then experts can interpret.

Finally, several VA tools have been developed for specific domains of research, such as medicine [HFM<sup>\*</sup>10, VFB<sup>\*</sup>08, NVKS14, CBR<sup>\*</sup>08], biology [AOD<sup>\*</sup>19, SLL<sup>\*</sup>14], security [AZV<sup>\*</sup>16], and social sciences [MKAN13]. However, VISRULER is a model-agnostic solution that could be modified to work with various domains, depending on the given data set and the domain expert.

### 3. Target Groups and Design Goals

In the InfoVis/VA communities, most of the research in explainable ML focuses on assisting *ML experts and developers* in understand-

ing, debugging, refining, and comparing ML models [CMJK20, CMJ<sup>\*</sup>20]. In this paper, we expand our method to involve another target group: the various *domain experts* affected by the ML progress in fields such as finance, social care, and health care. With the growing adoption of ML in different areas, domain experts with little knowledge of ML algorithms might still want (or be required) to use them to assist in their decision-making. On the one hand, their trust in such decisions could be low due to a lack of in-depth knowledge on how models are learning from the training data. On the other hand, ML experts often have little prior knowledge about the data from particular domains. Thus, the primary goal of VISRULER is to combine the best of both worlds, i.e., to offer a solution that combines the benefits from both expert groups.

Our design goals (**G1–G5**) originate from the analysis of the related work in Section 2, especially the three design goals from Zhao et al. [ZWLC19] and the four questions from Ming et al. [MQB19]. Also, our experience from the development of VA tools [CMKK21a, CMKK21b] for constructing powerful and diverse ML ensembles played a vital role. The implementation of the following design goals is described in Section 4.

**G1: Comparison of performance and architecture of models for selecting the most effective ones.** The comparison between models should be supported with various measurements, as follows: (1) illustrate the performance of each model based on multiple validation metrics; (2) distill the number of false-positive and false-negative instances from the confusion matrix for every model; and (3) derive the number of decision trees and decision paths (or simply *decisions*) per model, to compare their structure.

**G2: Investigation of the contribution of global features according to different models and algorithms.** Following the preceding goal, users should be guided through the process of selecting important features. Thus, it is crucial to enable the comparison between per-algorithm and per-model feature importances.

**G3: Exploration of alternative clusters of decisions for global explanation and case-based reasoning.** The summarization of the decisions in a single view that combines the decisions of different algorithms and models should be accomplished to allow users to assess the influence of each decision. For example, some decisions could overfit, and others could contain a mixture of instances falling in different classes. This last phenomenon increases their impurity. Users should be able to interact and explore this decision space.

**G4: Comparison of decision rules based on local feature ranking.** The global features described in G2 might not be similarly important for specific decisions, hence, local feature ranking via contrastive analysis [ZHPA13] could shed some light upon this task. Moreover, the interpretation of rules extracted from the space of solutions (see G3) could be achieved if users are capable of investigating the values of both training and testing instances.

**G5: Identification of the different types of failure cases and confrontation via manual decisions.** Failure to converge to a certain result due to the disagreement of the ML models should be highlighted to users. For instance, if there is no uniformity in the final decision or the majority voted for the wrong result, then it could be that these instances are outliers, borderline cases, or completely misclassified; being able to explore such cases easily is essential.

#### 4. VisRuler: System Overview and Use Case

To accomplish the aforementioned design goals, we have developed VISRULER. The backend of our VA tool was built using Python, Scikit-learn [PVG\*11], and Flask [Fla10]. As for the frontend, we utilize JavaScript, Vue [vue14], D3 [D311], and Plotly.js [plo10].

The tool consists of five main interactive visualization panels (Figure 1): (a) models overview (G1), (b) global feature ranking (G2), (c) decisions space (G3), (d) manual decisions (G4), and (e) decisions evaluation (G5). Our proposed **workflow** is a two-party system with the ML expert on the one side and the domain expert on the other (see Figure 2). The above-mentioned panels of our tool support the experts’ collaborative effort, specifically: (i) the ML expert should select powerful and diverse models from the two separate algorithms based on their performance assessed by validation metrics (Figure 1(a)); (ii) during this phase, the ML expert should choose which features are important for the active models compared to all models (see Figure 1(b)); (iii) in the next exploration phase, both experts should examine which decisions explain the data set globally and decide upon impactful decisions for a specific test instance (cf. Figure 1(c)); (iv) in this same phase, the domain expert should interpret the manual decisions selected in order to gain insights about the models’ decisions—either globally or locally—for a particular test instance (Figure 1(d)); and (v) in the final phase, the domain expert can evaluate the agreement and extract suitable manual decisions while the ML expert should search for new models if the search did not reach a satisfactory level according to the domain expert (Figure 1(e)). Overall, this is an iterative process with a final goal to receive insightful decisions that should be interpretable for all counterparts. Details about the different views within the panels can be found below.

The workflow of VISRULER is model-agnostic as far as rules can be extracted from the deployed ML algorithms. Currently, the implementation uses two rather popular EL methods: (1) RF and (2) AB (cf. green and blue colors in Figure 2, respectively). This choice was made intentionally because bagging methods work differently than boosting, as explained in Section 1. Furthermore, each data set is split in a stratified fashion (i.e., keeping the class balance in training/testing split) into 90% of training samples, and the remaining 10% becomes the test set. We also validate our results with cross-validation using 3-folds on the training set, and we scan the hyperparameter space for 10 iterations using Random search [BB12] in each algorithm separately. The common hyperparameters for both ML algorithms we experimented with (and their intervals) are: number of trees/estimators (2–20), maximum depth of a tree (10–25), and minimum samples in each leaf of a tree (1–10). An extra hyperparameter of RF is the maximum number of features to consider when looking for the best split ( $(\sqrt{\text{number\_of\_features}}) - (\text{number\_of\_features} - 1)$ ). AB has the learning rate (0.1–0.4).

In the following subsections, we explain VISRULER by describing a use case with the *World Happiness Report 2019* [JFRJD19] data set obtained from the Kaggle repository [Kag19]. This data set contains 156 countries (i.e., instances) ranked according to an index representing how happy the citizens of each country are. The six other variables that could be considered as features are: (1) *GDP per capita*, (2) *social support*, (3) *healthy life expectancy*, (4) *freedom to make life choices*, (5) *generosity*, and (6) *corrup-*

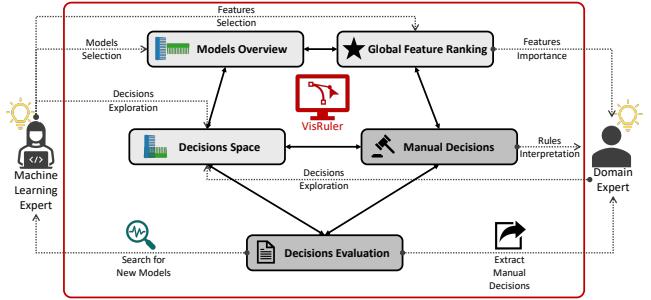


Figure 2: The VISRULER workflow allows the ML expert to select performant and diverse models, choose important features, investigate hyperparameters, and retrain models. The domain expert can explore robust decisions, compare them to global standards, identify local decisions for a specific test instance, and extract them.

tion perception. Because this data set does not contain any categorical class labels, we follow the same approach as in Neto and Paulovich [NP21b] to discretize the happiness score in three different bins. Hence, we are converting this regression problem into a multi-class classification problem [SK12]. Also in our case, the original variable Score becomes the target variable that our ML models should predict. In detail, the HS-Level-3 class contains 42 countries with happiness scores (HS) ranging from 6.13 to 7.76, the HS-Level-2 groups 79 countries from 4.49 to 6.13, and the HS-Level-1 class encloses 35 countries from 2.85 to 4.49.

#### 4.1. Models Overview

The exploration starts with an overview of how 10 RF and 10 AB models performed based on three validation metrics: accuracy, precision, and recall. The models are initially sorted according to the overall score, which is the average sum of the three metrics. Green is used for the RF algorithm, while blue is for AB. All visual representations share the same x-axis: the identification (ID) number of each model. The line chart in Figure 1(a) presents the worst to best models from left to right. The y-axis denotes the score for each metric as a percentage, with distinct symbols used for the different metrics. The Sankey diagram in Figure 1(a) visually maps a confusion matrix of only false-positive and false-negative values for each model, divided into two groups reflecting the two algorithms. It presents the confusion compared to all individual classes, as illustrated in both Figure 1(a) and Figure 3(a). The height of the lines indicates the increase or decrease in confusion from one model to the other sequentially, so the smaller the height of a line, the better a model’s prediction compared to the predecessor or successor. The same effect applies to each node that absorbs the lines. The bar charts in Figure 1(a) showcase the two main architectural components of the bagged and boosted decisions trees, which are the *number of trees/estimators* hyperparameter and the number of decisions generated from these trees for every model mapped in the y-axes, respectively. These visualizations allow users to check the structure of the individual models in a juxtaposed manner since the number of decisions is related to the number of trees and the maximum allowed depth of each tree (i.e., *max\_depth* hyperparameter).

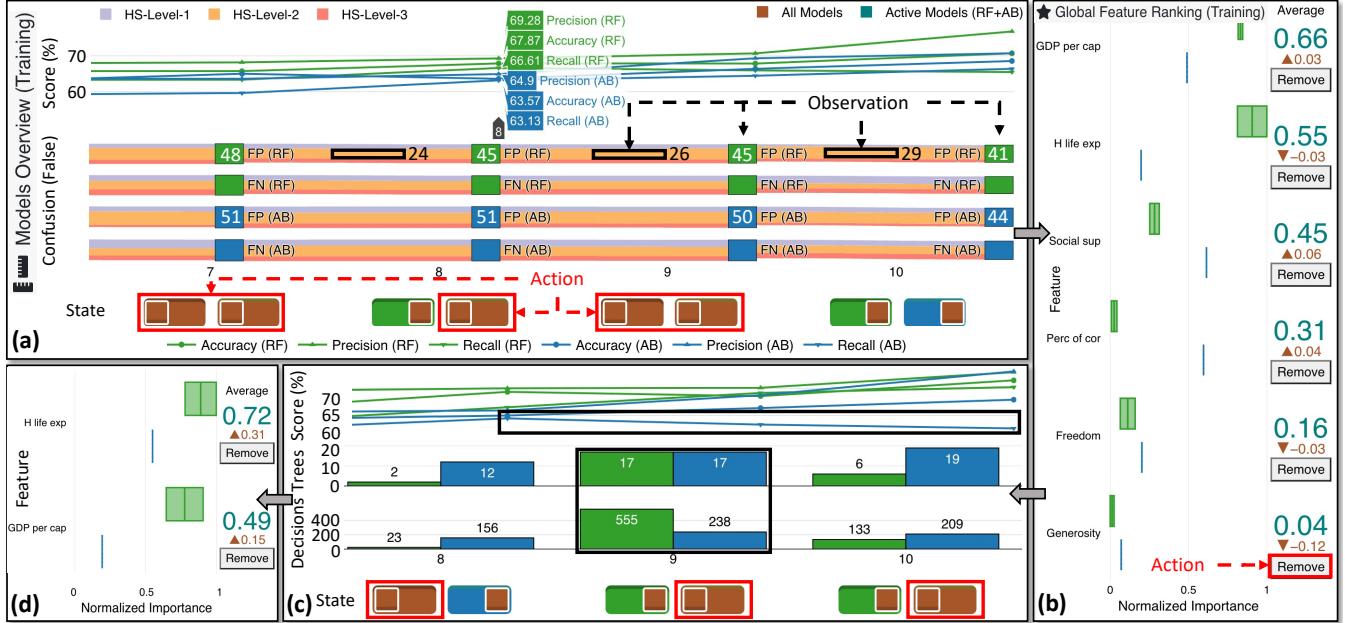


Figure 3: Exploration of ML models with VISRULER. View (a) presents the deactivation of all models except for RF8, RF10, and AB10, after careful consideration of their performance based on plentiful metrics displayed in the visualizations. If we look at (b), *Generosity* is the least important feature for the three active ML models, and particularly, its importance decreased while we deactivated most of the available ML models (see brown color). (c) indicates that after the retraining with 5 out of the 6 original features, the new AB8 is better than the subsequent models due to the decline in recall; AB8, RF9, and RF10 remain the only active models after this step. In the box plot in view (d), *H life exp* becomes the most important feature by far than *GDP per cap*. Thus, these features swapped places compared to view (b).

Finally, the state shown in Figure 1(a) designates which models are currently active (green or blue, respectively). In order to enable the comparison between the currently active model against all models, each icon for an active model contains a brown-colored slider thumb (Figure 1(a), including the legend in the top-right corner).

In our use case, we observe that models with ID number 8 and above slightly outperform the rest; notably, recall in AB7 is much lower than AB8 and beyond (cf. Figure 3(a), line chart). While RF models perform consistently better than AB models, as shown in both the line chart and the Sankey diagram of Figure 3(a), there is an improvement in the score of AB10. Therefore, we decide to keep only this model. Furthermore, since RF8 is more reliable in training instances for the HS-Level-2 class due to false-positives being lower than the equivalent for RF9 and RF10 (Figure 3(a), Sankey diagram), we keep this model and RF10, i.e., the top-performing model of the RF algorithm. In consequence, RF8, RF10, and AB10 are active models after selecting the corresponding states.

#### 4.2. Global Feature Ranking

The box plots which aggregate per-algorithm importance (see Figure 1(b)) provide a holistic view of the performance of the models. Each pair of boxes is related to a unique feature, summarizing the active models' normalized importance per feature (from 0 to 1, i.e., worst to best). The box plots are sorted according to the average values of all active models, which is visible as a number in teal. The difference to all models being active is evident with arrows facing up for increase or down for decrease in per-feature importance.

At this point, we want to investigate which features of the training set impacted the predictions more (see Figure 3). Interestingly, *GDP per cap*, *H life exp*, and *Social sup* are the top three features in the general ranking, as in [NP21b]. A surprising outcome is that, although two of the features mentioned above are still the most important for the selected RF models (all except *Social sup*), this is not true for the AB model. As seen in Figure 3(b), *Social sup*, *Perc of cor*, and *GDP per cap* are vital features for the AB algorithm in general. This pattern supports our hypothesis that different algorithms might take into account alternative features and should be combined to provide a holistic view. On the contrary, *Generosity* is unimportant for all models, specifically for the active models, since there is a  $-0.12$  decrease in importance. Thus, we choose to remove this feature and retrain without it (cf. Figure 3(b)). For the RF algorithm (green), we pick the most performant models based on the overall score (Figure 3(c)), rightmost models). However, AB8 is better overall than the subsequent AB models due to the stable and high recall value (Figure 3(c), line chart). In a one-to-one comparison between RF9 and AB9 with the bar charts, we recognize that while they have the same *number of estimators* (i.e., 17 trees), the two models produce 555 and 238 decisions, respectively. In this case, bagged decision trees allow a higher maximum depth than the equivalent boosted decision trees. After the selection of the new models, the most important features collectively are *H life exp* with 0.72 and *GDP per cap* with 0.49, as illustrated in Figure 3(d); the opposite was valid in Figure 3(b). The new AB model considers the same features more important as the RF models. After this phase is over, AB8, RF9, and RF10 are the remaining three active models.

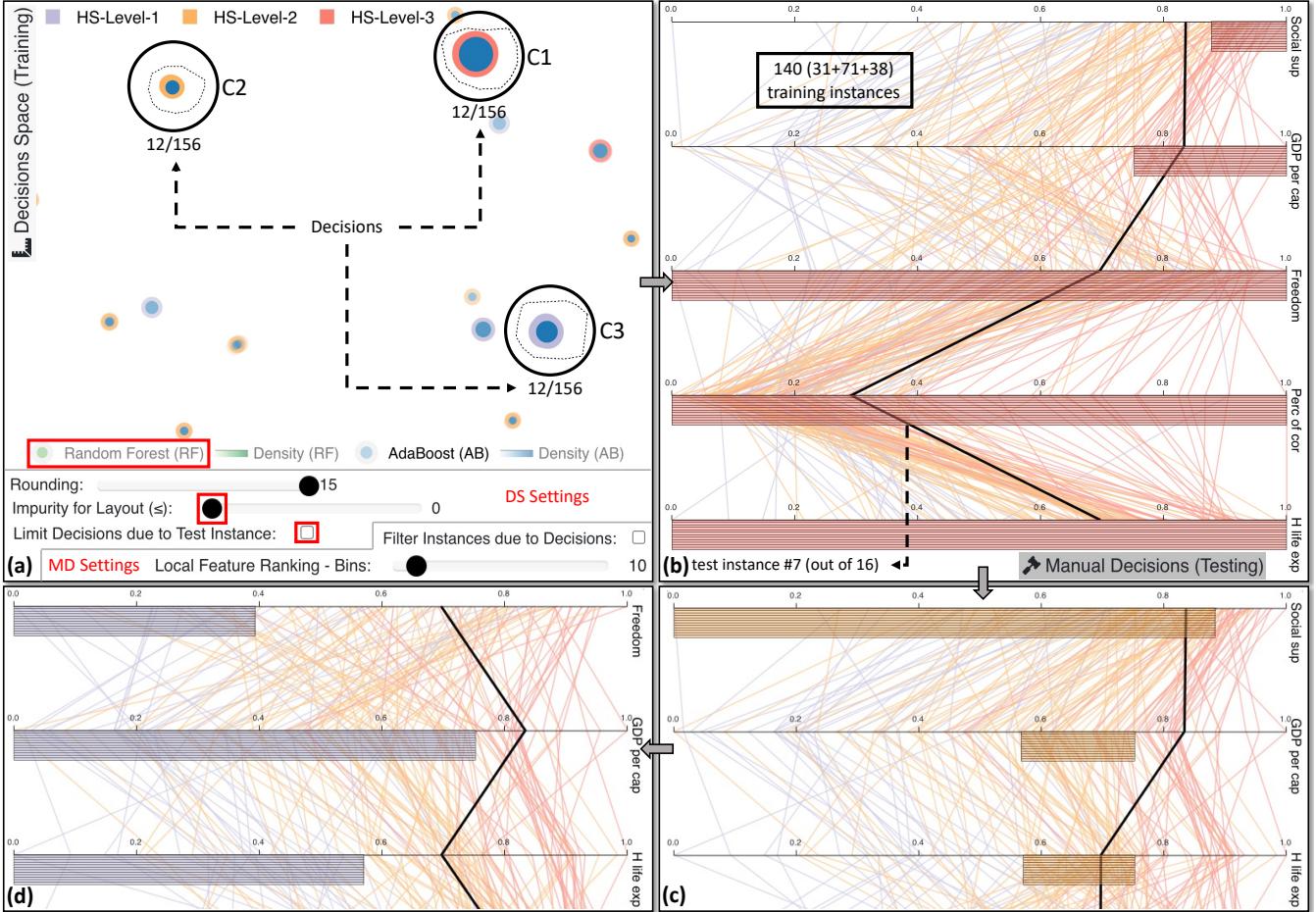


Figure 4: Examining several pure global decisions from the active AB model. In (a), we select step-by-step three clusters of 12 identical decisions each. Note that this screenshot is composed of the Decisions Space (DS) view and the settings for the same view plus the settings for the Manual Decisions (MD) view. The decisions for  $\textcircled{1}$  classify training instances only for HS-Level-3 class (as depicted in (b)). Similarly,  $\textcircled{2}$  contains decisions for HS-Level-2 (visible in (c)), while  $\textcircled{3}$  for the remaining class, as shown in (d). The 7<sup>th</sup> test instance, which is currently under investigation, cannot be classified by those prior decisions. However, it most likely belongs in the medium- or the high-level class.

#### 4.3. Decisions Space

The projection-based view in Figure 1(c) is produced by using UMAP [MHM18] with variable  $n\_neighbors$  hyperparameter and  $min\_dist$  set to 0.1. In the visual embedding, decisions are clustered based on their similarity according to the ranges they comprise for each feature, as in [ZWLC19]. To determine the optimal number of clusters to be visualized, DBSCAN [EKSX96] is used to compute an estimated number of core clusters from the derived decisions for a data set, which is then used to tune the  $n\_neighbors$ , with a minimum of 2 and a maximum of 100 neighbors (the aim is to have the same magnitude in both). The green color in the center of a point indicates that a decision is from RF, while blue is for AB. The outline color exposes the training instances' class based on a decision's prediction. The size maps the number of training instances that are classified by a specific decision, and the opacity encodes the impurity of each decision. Low impurity (with only a few training instances from other classes) makes the points more opaque. The positioning of the points can be useful to observe if both RF and

AB models produced similar rules, offering a comparison between algorithm decisions. The histogram in Figure 1(c) shows the number of decisions (y-axis) and the distribution of training instances in these paths (x-axis), and can also be used to filter the number of visible decisions to avoid overfitting rules containing only a few instances or general rules that might not apply in problematic cases.

Multiple interactions are possible in this view. The rounding slider (set to 15) allows users to round all decisions' range values to the desired decimal points. The comparison mode (active in Figure 1(c)) enables users to anchor groups of points and compare the selection against any other cluster. The two alternative choices are to present either the overlap or difference between the handpicked groups; the *Detach* button is for canceling this mode. Density views assist users in observing the distribution of RF against AB decisions in the projection, which is helpful if large amounts of decisions are visualized (see Supplemental Figure S1). The *Limit Decisions due to Test Instance* checkbox alters the layout and changes global decisions' exploration to local for a particular case. Finally, a limit

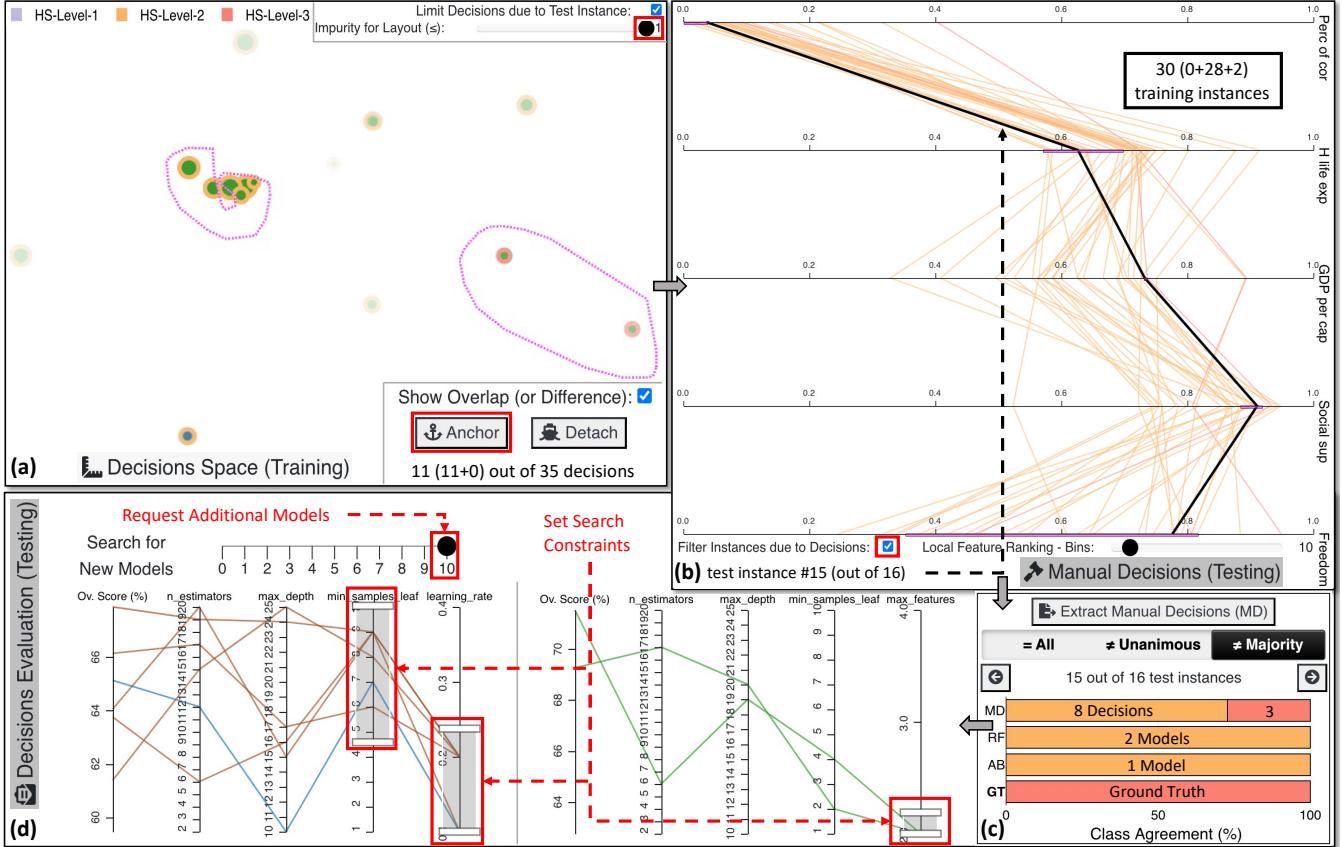


Figure 5: An outlier case exploration, the final prediction, and the training of another bunch of RF and AB models. (a) presents the anchoring of a cluster of 8 HS-Level-2 decisions to compare the overlapping rules against 3 HS-Level-3 decisions. In (b), after checking the common regions of agreement for the two clusters, we conclude that *Perc of cor* and *H life exp* are relatively low for the 15<sup>th</sup> test instance to belong in HS-Level-3 class. However, the other values for the remaining features are arguably rather high. In (c), we observe that all models voted for the average class while only the 3 selected manual decisions are supporting this case to be categorized as HS-Level-3 country. (d) showcases a potential search for new models by setting constraints in the hyperparameters according to the knowledge acquired from the initial training.

can be set for the acceptable impurity that is visible. If a decision is more impure than the currently chosen value, then it becomes almost transparent. As this view is tightly connected with the visualization of the following view, we proceed directly to Section 4.4.

#### 4.4. Manual Decisions

The vertical Parallel Coordinates Plot (PCP)-like view in Figure 1(d) illustrates the range values per feature for each selected decision (in this case, the comparison mode is active). The polylines represent the training instances and are color-encoded based on the ground truth (GT) class. There are two options here: either select to filter instances and show those that belong to the selected rules (see Figure 1(d)) or present all training instances at once (see Figure 4(b)–(d)). For example, in Figure 4(b), we see 12 identical rules that classify the training instances in the HS-Level-3 class (the red colored horizontal lines). The thick black polyline is the currently explorable test instance; users can compare it to the training instances that the models trained upon. All ranges for the features are normalized from 0.0 to 1.0. Scrolling is implemented when many

decisions must be shown or the number of features is large. The order of the features is initially the global one, as described in Section 4.2. When a group of points is selected using the lasso tool in the *Decisions Space*, a contrastive analysis [ZHPA13] is used to rank the features and help the user to find out unique features that explain a cluster's separation from the rest of the points. The computation works as follows: (1) break each feature into two disjoint distributions: the values inside the selected group vs. all the rest of the points; (2) discretize the two distributions of each feature into bins based on the *Local Feature Ranking - Bins* value set by the user (default is 10); (3) compute the cross-entropy [MPRO05] between the two distributions of each feature: higher values of cross-entropy suggest more unique features (i.e. the within-selection distribution is very different than the rest), while lower values suggest more common, shared features; and (4) rank the features based on step 3, with the more unique features nearer the top.

To investigate the global decisions based on the AB8 model we set the impurity to 0, disable limiting decisions based on the current test instance, and hide the RF models (cf. Figure 4(a)). We notice from the size of the decisions that if we analyze three core

clusters (c<sub>1</sub>)–(c<sub>3</sub>) we can get a better understanding of global decisions (see Figure 4(a)). In Figure 4(b), all 140 training instances (31 + 71 + 38 spread across the classes) are observable together with the 7<sup>th</sup> test instance, which is currently under investigation. From Figure 4(b), we see that *Social sup* and *GDP per cap* should be very high for test instances to belong to this class. In contrast, for test instances to be in the HS-Level-2 class, they need to have a low-to-average *Social sup*, and average *GDP per cap* and *H life exp* (Figure 4(c)). Low values in the features (1) *Freedom*, (2) *GDP per cap*, and (3) *H life exp* are common for the low score in happiness countries (see Figure 4(d)), as also identified by [NP21b]. Regarding Saudi Arabia (the 7<sup>th</sup> test instance), it does not appear to belong to any of those decisions, but it is far away from the values reported for the HS-Level-1 class. It has a very high *GDP per cap* to belong in the average class, but the *Social sup* is on the lower side. Despite that, *GDP per cap* is 1 out of the 2 most important features according to the analysis in Section 4.2. Our conclusion matches the fact that it was ranked in 28<sup>th</sup> place out of the 156 countries, thus, belonging to the list of 42 countries classified as HS-Level-3.

#### 4.5. Decisions Evaluation

The panel in Figure 1(e) contains interactive views that help users find outliers, borderline cases, and misclassified cases in the test set. The first main view allows users to extract the manual decisions (MD) selected in the previous phase (see Section 4.4). It also guides users in concentrating on cases where the majority of the RF and AB models disagreed when compared to the GT, or for models that did not vote unanimously. Furthermore, it is possible to go through all test instances one by one. The class agreement between RF and AB models, MD, and the GT is demonstrated via a horizontal stacked bar chart. The colors encode the different classes, and the length of each bar is the number of decisions for (1) MD, (2) RF models, (3) AB models, and (4) the GT (the latter always fills the entire bar). The second main view targets users that want to train new models based on the *Ov. Score (%)* of each previously-trained model. The two separate standard PCPs present the active RF models in green and the active AB models in blue, respectively. The brown color is used for the inactive models in both visualizations.

Checking the cases where the majority of the models disagree with the GT, we stop in the 15<sup>th</sup> test instance. Figure 5(a) shows the decisions applicable for this unusual case. We use the comparison mode to select a pure cluster on the left to juxtapose it with decisions classifying countries as HS-Level-3 on the right. Anchoring these clusters of points shows us the overlap of value ranges for the different features, as depicted in Figure 5(b). 28 out of the 30 training instances are similar to this test instance and belong to the HS-Level-2 class. The ranking of the features indicates that *Perc of cor* and *H life exp* are two unique features for the selected points, with low values for the former and average values for the latter, as in [NP21b]. Furthermore, for the first four features, the overlap is narrow between the two selected clusters, indicating that this instance could be considered an outlier. Indeed, Figure 5(c) presents that 8 out of the 11 decisions consider this instance as HS-Level-2. All active models are wrongly predicting Trinidad and Tobago (i.e., the 15<sup>th</sup> test instance) as an average HS country. Interestingly, the 3 MD of the RF models classified this country as HS-Level-3.

From the analyses in the previous subsections and the overall score of the RF and AB models, we observe that the most performant models for RF consider only 2 features when splitting the nodes (i.e., *max\_features* hyperparameter). The PCPs in Figure 5(d) enable us to scan the internal regions of the hyperparameters' solution space for RF. As for AB, the *learning\_rate* should be as low as possible for this specific data set, as seen in Figure 5(d). Also, by searching for models with high values for *min\_samples\_leaf*, AB models are created with complex decision trees compared to simple decision stumps, which seems to be an appropriate limitation of the hyperparameter space that could lead to better models. After all these constraints, we move the *Search for New Models* slider from 0 to 10 in Figure 5(d) to request 10 additional models for each algorithm with the hope of discovering more powerful ones.

#### 5. Usage Scenario

In this section, we describe a hypothetical usage scenario with a collaboration of a model developer (Amy, the ML expert) and a bank manager (Joe, the domain expert) who handles granting loans to customers. Joe wants to use VISRULER to improve the evaluation process of loan requests, so he asks Amy to use VISRULER to train ML models based on a data set collected over years of accepting or rejecting loans in the bank. The data set includes 1,000 instances/customers and 9 features/customer information, with 300 rejected (purple) and 700 accepted (orange) applications. This data set is, in reality, a pre-processed version [ZWLC19,NP21a] of German Credit Data from the UCI ML repository [DG17].

**Exploration and Selection of Algorithms and Models.** Following the workflow in Section 4, Amy loads the data set and checks the score of each model based on the three validation metrics (Figure 1(a)). For the AB algorithm, in blue, all models have a relatively low value for the recall metric, except for AB8. Also, AB7 performs very well for the Accepted class (orange), since the false-negative (FN) line reduces in height compared to all other models. Therefore, she decides to keep only AB7 and AB8. By looking at the Sankey diagram in Figure 1(a), Amy infers that RF4 and RF5 are the two models with low confusion, due to only 135 false-positive (FP) instances. She picks RF5 because it is the subsequent model from RF4, which means that the overall score is slightly higher. The top RF models on the right-hand side also caught her attention, with RF9 and RF10 being the best options. She thinks that either of them could do the job, as they appear redundant due to similar confusion and values in both the Sankey diagram and the line chart (cf. Figure 1(a)). The bar charts below—which highlight the difference in the architectures of these RF models—help her to choose: with only 7 decision trees and 589 decision paths (compared to 18 and 1,483), RF9 is simpler. She concludes that RF9's simplicity will make Joe's exploration of decisions more manageable at a later phase. Consequently, she deactivates RF10 and continues the feature contribution analysis with RF5, RF9, AB7, and AB8 models.

**Examining the Global Contribution of Features.** After this new selection of models, Amy observes in Figure 1(b) that most features (except for the last two) are more important now than in the initial state. *Ins\_perc* and *Val\_sa\_st* importances drop only by 0.01, implying these features are stable. She suggests Joe to keep all

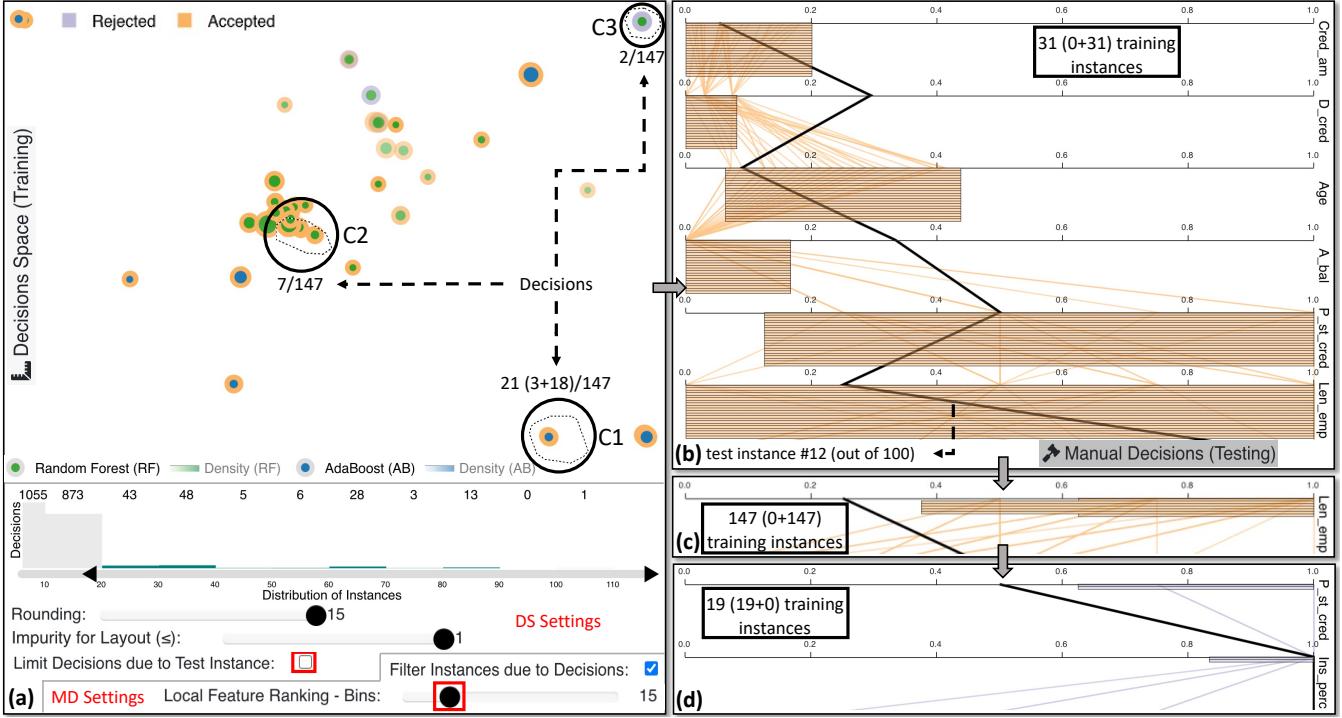


Figure 6: The exploration of clusters of decision paths from both ML algorithms. View (a) presents the selection of three clusters of global decisions that classify multiple training instances, thus, avoiding unimportant paths that might overfit. (b) provides an in-depth analysis of the decisions rules affected by  $\textcircled{2}$ . In (c),  $\textit{Len\_emp}$  emerges as a unique feature that characterizes  $\textcircled{2}$  with values from approximately 0.4 to 1.0. Finally in (d), high values in  $P_{st\_cred}$  and  $Ins\_{perc}$  turn over the prediction of the applicant to reject, visible via the exploration of  $\textcircled{3}$ .

features for now and explore the differences through the decision rules later on. Another interesting insight is that  $A_{bal}$  is the most important feature for the RF models, while the AB models prefer  $D_{cred}$  (see Figure 1(b)). This could indicate that mixing models' decisions from different algorithms is beneficial.

**Explanations through Global Decision Rules.** Joe starts his exploration by examining the global decision rules that can help him make accurate decisions for specific cases in the future. He focuses on the 12<sup>th</sup> test instance, which is a customer application reviewed by a colleague, Silvia (cf. usage scenario by Neto and Paulovich [NP21a]). First, he unchecks *limiting the decisions due to the test instance*, as illustrated in Figure 6(a). At this point, Amy identifies several decisions that classify only fewer than 20 customers; she thinks: “these are not so generic after all”. Indeed, the larger the number of instances classified by one rule, the more generic and important it is (if the impurity is low). Consequently, they decide to increase the lower boundary of decisions, filtering out 1,928 decisions (see Figure 6(a), bar chart). After the update, Joe focuses on the UMAP [MHM18] projection. He observes multiple groups of points that could be worthy of further investigation. He selects a couple of samples from different areas, e.g.,  $\textcircled{1}$  with 3 RF and 18 AB decisions. Another cluster with 7 decisions is  $\textcircled{2}$  that solely predicts accepted loan applications. On the contrary,  $\textcircled{3}$  contains 2 pure decisions (due to high opacity) that produce rules which reject loans. Joe increases the *discretization of local feature ranking* from 10 to 15 bins to raise the sensitivity of difference between decision rule ranges, and he *filters the instances due to the*

*decisions* to observe clearer trends. From Figure 6(b), Joe recognizes that  $\textcircled{2}$  decisions are all identical, having the same ranges for every feature. Also, he understands that low *credited amount* ( $Cred\_am$ ) and short *duration of credit* ( $D_{cred}$ ) are essential factors for accepting a loan application. *Account balance* is also vital because all loans are accepted when there is no account ( $A_{bal}$  being 0). Figure 6(c) reveals another intriguing pattern, that is, *the length of current employment* should be average to extremely high (from approximately 0.4 or 0.6 and above) for applications to get accepted. In contrast, Figure 6(d) presents that if *payment status of previous credit* ( $P_{st\_cred}$ ) and *instalment per cent* ( $Ins\_{perc}$ ) are relatively high, the applications were rejected. The 12<sup>th</sup> customer has an account without any balance, and the  $D_{cred}$  is relatively high, which flips the prediction toward rejection. Luckily, Silvia also provided an adequate justification to the customer [NP21a].

**Extracting Manual Decisions through Local Investigations.** At this point Joe knows and understands the main decision rules, but a new customer arrives. Focusing on the decisions for this case (i.e., 90<sup>th</sup> test instance), he sets impurity to less than 0.3 (cf. Figure 1(c), slider) to make impure decisions more transparent. Two fairly pure decisions from RF5 (visible due to hovering) and RF9 contradict each other. Joe uses the comparison mode, anchors 1 out of the 2 decisions, and selects the other with the lasso tool. The comparison in Figure 1(d) designates that 8 similar customers' applications were rejected while 12 were accepted. The small overlap in  $Cred\_am$ ,  $D_{cred}$ , and  $Age$  suggest that this is a borderline case.  $Cred\_am$  seems a bit arbitrary for the training data since only

a small amount of applications in-between accepted applications were rejected, see Figure 1(d), feature on top. However, a clear insight is that if  $D_{cred}$  was lower, the application should have been accepted, while the opposite effect is true if the *duration of credit* increases. Unexpectedly, RF models vote for accepting this loan application while AB models reject (cf. Figure 1(e), top view). Besides that, the manual decisions are also in-between the two classes, which further enhances Joe’s assumption that this is a borderline case. As AB models propose rejection and RF9 produces a decision for rejecting this application, he follows these recommendations. Nonetheless, Joe asks Amy to search and train new performant ML models (see next paragraph).

#### Tuning the Search for Bagged and Boosted Decision Trees.

Amy sees two possibilities of improvement for the RF in Figure 1(e), bottom view. One is to limit the *max\_features* to 7 and 8 because they produced the best models so far. The second strategy is to pick 3 and 4 for the same hyperparameter to explore an entirely new space of currently unexplored models. Basically, she believes it is better to try both strategies in two separate runs. As for the AB, she reasons that selecting 0.1 and 0.2 for the *learning\_rate* is a wise choice. Although it may take more time to retrain the AB models, they probably will be more powerful than with the other setting due to historical data. She performs the above actions, and finally, another cycle of exploration is unfolded for both experts.

## 6. User Study

We conducted a user study to evaluate our tool’s effectiveness in supporting decision-making based on many alternative decision paths. As in prior works [MQB19, NP21a], we created five questions (Qs) that cover VISRULER’s different views, focusing on appraising the goals described in Section 3 with the use case outlined in Section 4 as the GT (see Supplemental Table S2).

**Demographics and Instructions.** 7 male and 5 female volunteers aged 23 to 49 (mean:  $\approx 33$ ) participated in our study, all with at least an MSc degree (and 2 PhD’s). None of them knew the data set used, and no colorblindness issues were reported. 4 of the participants were highly knowledgeable in visualization and 7 in ML, while the rest had limited knowledge regarding all aspects and 4 of them had never worked with any EL method. The initial step of the study was to watch an  $\approx 18$ -minute video tutorial about bagging and boosting concepts, VISRULER’s goals, and how to work with our tool to analyze decision paths, using the Iris data set [Fis36]. The participants experimented for five minutes with Iris, and then proceeded to use the data set described in Section 4. They were asked to answer five questions (cf. Supplemental Document S3) and provide qualitative feedback via the ICE-T questionnaire [WAM\*19].

**Question-related Results.** After the initial setting shown in Figure 3(a), all participants decided to exclude *Generosity* in Q1, which happened in 2.03 minutes on average. For Q2, 9 participants followed our GT, as described in Figure 3(c). The remaining attendees selected AB10 instead of AB8. This action led to 5 test instances in conflict compared to 3 in our analysis (Figure 5(c) presents a single case). This result could be a strong indication that our approach is essential for making such decisions. To respond in Q2 and Q3, participants took 4.04 and 2.58 minutes on average,

Table 1: Analyzed results from the ICE-T feedback [WAM\*19].

| Components     | Insight         | Time            | Essence         | Confidence      | Average         |
|----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Participant 1  | 6.63            | 6.80            | 6.50            | 7.00            | 6.73            |
| Participant 12 | 6.75            | 6.40            | 7.00            | 6.75            | 6.73            |
| Participant 11 | 7.00            | 6.80            | 7.00            | 5.50            | 6.58            |
| Participant 8  | 6.88            | 6.80            | 6.75            | 5.25            | 6.42            |
| Participant 9  | 6.00            | 7.00            | 6.50            | 6.00            | 6.38            |
| Participant 3  | 6.88            | 6.60            | 6.50            | 5.25            | 6.31            |
| Participant 10 | 6.25            | 6.60            | 6.25            | 6.00            | 6.28            |
| Participant 6  | 6.50            | 6.00            | 6.50            | 6.00            | 6.25            |
| Participant 5  | 6.38            | 6.00            | 6.50            | 6.00            | 6.22            |
| Participant 2  | 5.63            | 6.20            | 5.75            | 6.00            | 5.89            |
| Participant 7  | 5.63            | 5.00            | 5.75            | 6.00            | 5.59            |
| Participant 4  | 5.75            | 5.20            | 6.25            | 4.67            | 5.47            |
| 95% C.I.       | $6.35 \pm 0.32$ | $6.28 \pm 0.41$ | $6.44 \pm 0.25$ | $5.87 \pm 0.41$ | $6.24 \pm 0.26$ |

Legend:  
7  
6  
5  
4  
3  
2  
1

respectively. The most time-consuming question was Q4 with an average response time of 6.15 minutes (but with very accurate results, see Figure 4(b)). The average time taken for Q5 was 6.07 minutes, with only one wrong answer (Figure 5(b)).

**Qualitative Results.** In Table 1, the mean scores of each component of the ICE-T form [WAM\*19] for every participant are displayed along with the two-tailed 95% confidence intervals (CIs) per component ( $t^* = 2.201, N = 12$ ). Higher values in green designate good results, as opposed to red. VISRULER has received a few 7.0 scores, and most are at least 6.0 and above (the lowest score is 4.67). Essence, Insight, and Time received a large score which means users found our tool competent in portraying decisions, guiding users to come up with fundamental questions, and performing these discoveries quickly. The Confidence was lower, with a mean value of 5.87. However, this value still makes VISRULER a reliable and trustworthy VA tool based on Wall et al. [WAM\*19].

## 7. Discussion and Conclusions

We presented VISRULER, a VA tool that allows users to explore diverse rules extracted from bagged and boosted decision trees to reach a consensus about a final decision for each individual case. The multiple coordinated views facilitate the selection of diverse and performant models, the characterization of per-feature contribution, the management of multiple decisions, the analysis of global decisions, and support case-based reasoning. Finally, we validated the usability and efficacy of VISRULER via a user study.

**Limitations.** Although VISRULER can visualize thousands of decision paths verified by the usage scenario in Section 5, the cluttering of the dimension reduction methods could be deemed as an intrinsic difficulty. Also, efficiency might be problematic if numerous models are simultaneously active and produce too many decisions. In such cases, the vertical PCP may be challenging to interpret because it requires users to scroll through a list of decisions that expands by the number of features. Another limitation is the extensive (but unavoidable) use of color that might hinder our tool from operating with more than a few classes. All these limitations indicate future directions for our work.

## Acknowledgements

This work was partially supported through the ELLIIT environment for strategic research in Sweden.

## References

- [AEK00] ANKERST M., ESTER M., KRIESEL H.-P.: Towards an effective cooperation of the user and the computer for classification. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2000), KDD '00, Association for Computing Machinery, p. 179–188. [doi:10.1145/347090.347124](#). 3
- [AOD\*19] ABRAMOV D., OTTO J., DUBEY M., ARTANEGARA C., BOUTILLIER P., FONTANA W., FORBES A. G.: RuleVis: Constructing patterns and rules for rule-based models. In *2019 IEEE Visualization Conference (VIS)* (2019), pp. 191–195. [doi:10.1109/VISUAL.2019.8933596](#). 3
- [AZV\*16] AUPETIT M., ZHAUNIAROVICH Y., VASILIADIS G., DACIER M., BOSHMAF Y.: Visualization of actionable knowledge to mitigate DRDoS attacks. In *2016 IEEE Symposium on Visualization for Cyber Security (VizSec)* (2016), pp. 1–8. [doi:10.1109/VIZSEC.2016.7739577](#). 3
- [BB12] BERGSTRA J., BENGIO Y.: Random search for hyper-parameter optimization. *Journal of Machine Learning Research* 13 (Feb. 2012), 281–305. [doi:10.5555/2188385.2188395](#). 4
- [BKSS14] BEHRISCH M., KORKMAZ F., SHAO L., SCHRECK T.: Feedback-driven interactive exploration of large multidimensional data supported by visual classifier. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)* (2014), pp. 43–52. [doi:10.1109/VAST.2014.7042480](#). 3
- [BN01] BARLOW T., NEVILLE P.: Case study: Visualization for decision tree analysis in data mining. In *IEEE Symposium on Information Visualization, 2001. INFOVIS 2001*. (2001), pp. 149–152. [doi:10.1109/INFVIS.2001.963292](#). 3
- [Bre96] BREIMAN L.: Stacked regressions. *Machine learning* 24, 1 (1996), 49–64. 2
- [Bre01a] BREIMAN L.: Random forests. *Machine Learning* 45 (Oct. 2001), 5–32. [doi:10.1023/A:1010933404324](#). 2
- [Bre01b] BREIMAN L.: Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science* 16, 3 (2001), 199 – 231. [doi:10.1214/ss/1009213726](#). 2
- [BvLH\*11] BREMM S., VON LANDESBERGER T., HESS M., SCHRECK T., WEIL P., HAMACHER K.: Interactive visual comparison of multiple trees. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)* (2011), pp. 31–40. [doi:10.1109/VAST.2011.6102439](#). 3
- [C19] CAVALLO M., C. D.: Clustrophile 2: Guided visual clustering analysis. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 267–276. [doi:10.1109/TVCG.2018.2864477](#). 3
- [CB20] CAO F., BROWN E. T.: Dril: Descriptive rules by interactive learning. In *2020 IEEE Visualization Conference (VIS)* (2020), pp. 256–260. [doi:10.1109/VIS47514.2020.00058](#). 3
- [CBR\*08] CARLSON J. M., BRUMME Z. L., ROUSSEAU C. M., BRUMME C. J., MATTHEWS P. C., KADIE C. M., MULLINS J. I., WALKER B. D., HARRIGAN P. R., GOULDER P. J. R., HECKERMAN D.: Phylogenetic dependency networks: Inferring patterns of CTL escape and codon covariation in HIV-1 Gag. *PLoS computational biology* 4, 11 (2008), e1000225. 3
- [CLG\*15] CARUANA R., LOU Y., GEHRKE J., KOCH P., STURM M., ELHADAD N.: Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2015), KDD '15, Association for Computing Machinery, p. 1721–1730. [doi:10.1145/2783258.2788613](#). 2
- [CMJ\*20] CHATZIMPARMPAS A., MARTINS R. M., JUSUFİ I., KUCHER K., ROSSI F., KERREN A.: The state of the art in enhancing trust in machine learning models with the use of visualizations. *Computer Graphics Forum* 39, 3 (June 2020), 713–756. [doi:10.1111/cgf.14034](#). 3
- [CMJK20] CHATZIMPARMPAS A., MARTINS R. M., JUSUFİ I., KERREN A.: A survey of surveys on the use of visualization for interpreting machine learning models. *Information Visualization* 19, 3 (July 2020), 207–233. [doi:10.1177/1473871620904671](#). 3
- [CMKK21a] CHATZIMPARMPAS A., MARTINS R. M., KUCHER K., KERREN A.: StackGenVis: Alignment of data, algorithms, and models for stacking ensemble learning using performance metrics. *IEEE Transactions on Visualization and Computer Graphics* (2021). [doi:10.1109/TVCG.2020.3030352](#). 3
- [CMKK21b] CHATZIMPARMPAS A., MARTINS R. M., KUCHER K., KERREN A.: VisEvol: Visual analytics to support hyperparameter search through evolutionary optimization. *Computer Graphics Forum* 40, 3 (2021), 201–214. [doi:10.1111/cgf.14300](#). 3
- [CPC19] CARVALHO D. V., PEREIRA E. M., CARDOSO J. S.: Machine learning interpretability: A survey on methods and metrics. *Electronics* 8, 8 (2019). [doi:10.3390/electronics8080832](#). 2
- [D311] D3 — Data-driven documents, 2011. Accessed December 2, 2021. URL: <https://d3js.org/>. 4
- [DB21] DENG J., BROWN E. T.: RISSAD: Rule-based interactive semi-supervised anomaly detection. In *EuroVis 2021 - Short Papers* (2021), The Eurographics Association. [doi:10.2312/evs.20211050](#). 3
- [DCB19] DI CASTRO F., BERTINI E.: Surrogate decision tree visualization interpreting and visualizing black-box classification models with surrogate decision tree. In *CEUR Workshop Proceedings* (2019), vol. 2327, CEUR-WS. 3
- [DG17] DUA D., GRAFF C.: UCI machine learning repository, 2017. Accessed December 2, 2021. URL: <http://archive.ics.uci.edu/ml>. 2, 8
- [DLH19] DU M., LIU N., HU X.: Techniques for interpretable machine learning. *Commun. ACM* 63, 1 (Dec. 2019), 68–77. [doi:10.1145/3359786](#). 2
- [Do07] DO T.-N.: Towards simple, easy to understand, an interactive decision tree algorithm. *College of Information Technology, Cantho University, Cantho, Vietnam, Technical Report* (2007), 06–01. 3
- [EAM14] EISEMANN M., ALBUQUERQUE G., MAGNOR M.: A nested hierarchy of localized scatterplots. In *2014 27th SIBGRAPI Conference on Graphics, Patterns and Images* (2014), pp. 80–86. [doi:10.1109/SIBGRAPI.2014.14](#). 3
- [EKSX96] ESTER M., KRIESEL H.-P., SANDER J., XU X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (1996), KDD'96, AAAI Press, p. 226–231. 6
- [FDCBA14] FERNÁNDEZ-DELGADO M., CERNADAS E., BARRO S., AMORIM D.: Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research* 15, 1 (Jan. 2014), 3133–3181. 2
- [Fis36] FISHER R. A.: The use of multiple measurements in taxonomic problems. *Annals of Eugenics* 7, 2 (1936), 179–188. [doi:10.1111/j.1469-1809.1936.tb02137.x](#). 10
- [Fla10] Flask — A micro web framework written in Python, 2010. Accessed December 2, 2021. URL: <https://flask.palletsprojects.com/>. 4
- [Fri01] FRIEDMAN J. H.: Greedy function approximation: A gradient boosting machine. *Annals of statistics* (2001), 1189–1232. 3
- [FS96] FREUND Y., SCHAPIRE R. E.: Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning* (San Francisco, CA, USA, 1996), ICML'96, Morgan Kaufmann Publishers Inc., p. 148–156. 2
- [FSA99] FREUND Y., SCHAPIRE R., ABE N.: A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence* 14, 5 (Sept. 1999), 771–780. 2

- [GGPPS13] GUERRA-GÓMEZ J., PACK M. L., PLAISANT C., SHNEIDERMAN B.: Visualizing change over time using dynamic hierarchies: TreeVerty2 and the StemView. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2566–2575. [doi:10.1109/TVCG.2013.231](https://doi.org/10.1109/TVCG.2013.231). 3
- [HC00] HAN J., CERCONE N.: RuleViz: A model for visualizing knowledge discovery process. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2000), KDD ’00, Association for Computing Machinery, p. 244–253. [doi:10.1145/347090.347139](https://doi.org/10.1145/347090.347139). 3
- [HFM\*10] HUMMELIN R., FERNANDES A. D., MACKLAIM J. M., DICKSON R. J., CHANGALUCHA J., GLOOR G. B., REID G.: Deep sequencing of the vaginal microbiota of women with HIV. *PLOS ONE* 5, 8 (08 2010), 1–9. [doi:10.1371/journal.pone.0012078](https://doi.org/10.1371/journal.pone.0012078). 3
- [HLLW19] HUANG Y., LIU Y., LI C., WANG C.: GBRTVis: Online analysis of gradient boosting regression tree. *J. Vis.* 22, 1 (Feb. 2019), 125–140. [doi:10.1007/s12650-018-0514-2](https://doi.org/10.1007/s12650-018-0514-2). 3
- [HTF01] HASTIE T., TIBSHIRANI R., FRIEDMAN J.: *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001. 2
- [JFRJD19] JOHN F. H., RICHARD L., JEFFREY D. S.: World happiness report 2019. *New York: Sustainable Development Solutions Network* (2019). 4
- [JLL\*20] JIA S., LIN P., LI Z., ZHANG J., LIU S.: Visualizing surrogate decision trees of convolutional neural networks. *Journal of Visualization* 23, 1 (2020), 141–156. 3
- [Kag19] World happiness report, 2019. Accessed December 2, 2021. URL: <https://www.kaggle.com/undsn/world-happiness>. 4
- [KCKS19] KOPITAR L., CILAR L., KOCBEK P., STIGLIC G.: Local vs. global interpretability of machine learning models in type 2 diabetes mellitus screening. In *Artificial Intelligence in Medicine: Knowledge Representation and Transparent and Explainable Systems* (Cham, 2019), Springer International Publishing, pp. 108–119. 2
- [KRS14] KIM B., RUDIN C., SHAH J.: The Bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2* (Cambridge, MA, USA, 2014), NIPS’14, MIT Press, p. 1952–1960. 2
- [KS08] KINGSFORD C., SALZBERG S. L.: What are decision trees? *Nature biotechnology* 26, 9 (2008), 1011–1013. 2
- [LBL16] 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* (New York, NY, USA, 2016), KDD ’16, Association for Computing Machinery, p. 1675–1684. [doi:10.1145/2939672.2939874](https://doi.org/10.1145/2939672.2939874). 2
- [Lip18] LIPTON Z. C.: The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue* 16, 3 (June 2018), 31–57. [doi:10.1145/3236386.3241340](https://doi.org/10.1145/3236386.3241340). 2
- [LJC16] LEE T., JOHNSON J., CHENG S.: An interactive machine learning framework, 2016. [arXiv:1610.05463](https://arxiv.org/abs/1610.05463). 3
- [LXL\*18] LIU S., XIAO J., LIU J., WANG X., WU J., ZHU J.: Visual diagnosis of tree boosting methods. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (Jan. 2018), 163–173. [doi:10.1109/TVCG.2017.2744378](https://doi.org/10.1109/TVCG.2017.2744378). 3
- [MGT\*03] MUNZNER T., GUIMBRETIÈRE F., TASIRAN S., ZHANG L., ZHOU Y.: TreeJuxtaposer: Scalable tree comparison using focus+context with guaranteed visibility. *ACM Trans. Graph.* 22, 3 (July 2003), 453–462. [doi:10.1145/882262.882291](https://doi.org/10.1145/882262.882291). 3
- [MHM18] MCINNES L., HEALY J., MELVILLE J.: UMAP: Uniform manifold approximation and projection for dimension reduction. *ArXiv e-prints* 1802.03426 (Feb. 2018). [arXiv:1802.03426](https://arxiv.org/abs/1802.03426). 6, 9
- [MJEP\*21] MARCÍLIO-JR W. E., ELER D. M., PAULOVICH F. V., RODRIGUES-JR J. F., ARTERO A. O.: ExplorerTree: A focus+context exploration approach for 2D embeddings. *Big Data Research* 25 (2021), 100239. [doi:10.1016/j.bdr.2021.100239](https://doi.org/10.1016/j.bdr.2021.100239). 3
- [MKAN13] MOUSSAÏD M., KÄMMER J. E., ANALYTIS P. P., NETH H.: Social influence and the collective dynamics of opinion formation. *PloS one* 8, 11 (2013), e78433. 3
- [MLMP18] MÜHLBACHER T., LINHARDT L., MÖLLER T., PIRINGER H.: TreePOD: Sensitivity-aware selection of Pareto-optimal decision trees. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018), 174–183. [doi:10.1109/TVCG.2017.2745158](https://doi.org/10.1109/TVCG.2017.2745158). 3
- [MPR05] MANNOR S., PELEG D., RUBINSTEIN R.: The cross entropy method for classification. In *Proceedings of the 22nd International Conference on Machine Learning* (New York, NY, USA, 2005), ICML ’05, Association for Computing Machinery, p. 561–568. [doi:10.1145/1102351.1102422](https://doi.org/10.1145/1102351.1102422). 7
- [MQB19] MING Y., QU H., BERTINI E.: RuleMatrix: Visualizing and understanding classifiers with rules. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 342–352. [doi:10.1109/TVCG.2018.2864812](https://doi.org/10.1109/TVCG.2018.2864812). 3, 10
- [NHS00] NGUYEN T., HO T., SHIMODAIRA H.: A visualization tool for interactive learning of large decision trees. In *Proceedings 12th IEEE Internationals Conference on Tools with Artificial Intelligence. ICTAI 2000* (2000), pp. 28–35. [doi:10.1109/TAI.2000.889842](https://doi.org/10.1109/TAI.2000.889842). 3
- [NP21a] NETO M. P., PAULOVICH F. V.: Explainable Matrix - Visualization for global and local interpretability of random forest classification ensembles. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2021), 1427–1437. [doi:10.1109/TVCG.2020.3030354](https://doi.org/10.1109/TVCG.2020.3030354). 2, 8, 9, 10
- [NP21b] NETO M. P., PAULOVICH F. V.: Multivariate data explanation by jumping emerging patterns visualization. *arXiv preprint arXiv:2106.11112* (2021). 2, 4, 5, 8
- [NVKS14] NIEMANN U., VÖLKZE H., KÜHN J.-P., SPILIOPOULOU M.: Learning and inspecting classification rules from longitudinal epidemiological data to identify predictive features on hepatic steatosis. *Expert Systems with Applications* 41, 11 (2014), 5405–5415. [doi:10.1016/j.eswa.2014.02.040](https://doi.org/10.1016/j.eswa.2014.02.040). 3
- [NWWH19] NSCH R. H., WIESNER P., WENDLER S., HELLWICH O.: Colorful Trees: Visualizing random forests for analysis and interpretation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)* (2019), pp. 294–302. [doi:10.1109/WACV.2019.00037](https://doi.org/10.1109/WACV.2019.00037). 2
- [OM99] OPITZ D., MACLIN R.: Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research* 11, 1 (July 1999), 169–198. 2
- [plo10] Plotly — JavaScript open source graphing library, 2010. Accessed December 2, 2021. URL: <https://plotly.com>. 4
- [PNWG17] PHILLIPS N. D., NETH H., WOIKE J. K., GAISSMAIER W.: FFTrees: A toolbox to create, visualize, and evaluate fast-and-frugal decision trees. *Judgment and Decision making* 12, 4 (2017), 344–368. 3
- [PSMD14] PADUA L., SCHULZE H., MATKOVIĆ K., DELRIEUX C.: Interactive exploration of parameter space in data mining: Comprehending the predictive quality of large decision tree collections. *Computers & Graphics* 41 (2014), 99–113. [doi:10.1016/j.cag.2014.02.004](https://doi.org/10.1016/j.cag.2014.02.004). 3
- [PVG\*11] PEDREGOSA F., VAROQUAUX G., GRAMFORT A., MICHEL V., THIRION B., GRISEL O., BLONDÉL M., PRETTENHOFER P., WEISS R., DUBOURG V., VANDERPLAS J., PASSOS A., COURNAPÉAU D., BRUCHER M., PERROT M., DUCHESNAY E.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12 (Nov. 2011), 2825–2830. [doi:10.5555/1953048.2078195](https://doi.org/10.5555/1953048.2078195). 4
- [RSG16] 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* (2016), KDD ’16, ACM, pp. 1135–1144. [doi:10.1145/2939672.2939778](https://doi.org/10.1145/2939672.2939778). 2

- [Sch90] SCHAPIRE R. E.: The strength of weak learnability. *Machine learning* 5, 2 (1990), 197–227. 2
- [SCS04] SONG H., CURRAN E. P., STERRITT R.: Multiple foci visualisation of large hierarchies with FlexTree. *Information Visualization* 3, 1 (2004), 19–35. doi:10.1057/palgrave.ivs.9500065. 3
- [SK12] SALMAN R., KECMAN V.: Regression as classification. In *2012 Proceedings of IEEE Southeastcon* (2012), pp. 1–6. doi:10.1109/SECon.2012.6196887. 4
- [SLL\*14] SYDOW J. F., LIPSMIEIER F., LARRAILLET V., HILGER M., MAUTZ B., MØLHØJ M., KUENTZER J., KLOSTERMANN S., SCHOCH J., VOELGER H. R., ET AL.: Structure-based prediction of asparagine and aspartate degradation sites in antibody variable regions. *PLoS one* 9, 6 (2014), e100736. 3
- [SMS\*21] STREEB D., METZ Y., SCHLEGEL U., SCHNEIDER B., ELASSADY M., NETH H., CHEN M., KEIM D.: Task-based visual interactive modeling: Decision trees and rule-based classifiers. *IEEE Transactions on Visualization and Computer Graphics* (2021), 1–1. doi:10.1109/TVCG.2020.3045560. 2
- [SR18] SAGI O., ROKACH L.: Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery* 8, 4 (July–Aug. 2018), e1249. doi:10.1002/widm.1249. 2
- [SYX\*20] SACHAN S., YANG J.-B., XU D.-L., BENAVIDES D. E., LI Y.: An explainable AI decision-support-system to automate loan underwriting. *Expert Systems with Applications* 144 (2020), 113100. doi:10.1016/j.eswa.2019.113100. 2
- [TKC17] TAM G. K. L., KOTHARI V., CHEN M.: An analysis of machine- and human-analytics in classification. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 71–80. doi:10.1109/TVCG.2016.2598829. 2
- [TM03a] TEOH S. T., MA K.-L.: PaintingClass: Interactive construction, visualization and exploration of decision trees. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2003), KDD ’03, Association for Computing Machinery, p. 667–672. doi:10.1145/956750.956837. 3
- [TM03b] TEOH S. T., MA K.-L.: StarClass: Interactive visual classification using star coordinates. In *Proceedings of the 2003 SIAM International Conference on Data Mining* (2003), SIAM, pp. 178–185. 3
- [vdEvW11] VAN DEN ELZEN S., VAN WIJK J. J.: BaobabView: Interactive construction and analysis of decision trees. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)* (2011), pp. 151–160. doi:10.1109/VAST.2011.6102453. 3
- [vdMH08] VAN DER MAATEN L., HINTON G.: Visualizing data using t-SNE. *Journal of Machine Learning Research* 9 (2008), 2579–2605. 3
- [VFB\*08] VIROS A., FRIDLYAND J., BAUER J., LASITHOTAKIS K., GARBE C., PINKEL D., BASTIAN B. C.: Improving melanoma classification by integrating genetic and morphologic features. *PLoS medicine* 5, 6 (2008), e120. 3
- [vue14] Vue.js — The progressive JavaScript framework, 2014. Accessed December 2, 2021. URL: <https://vuejs.org/>. 4
- [WAM\*19] WALL E., AGNIHOTRI M., MATZEN L., DIVIS K., HAASS M., ENDERT A., STASKO J.: A heuristic approach to value-driven evaluation of visualizations. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 491–500. doi:10.1109/TVCG.2018.2865146. 10
- [Wel19] WELLER A.: Transparency: motivations and challenges. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer, 2019, pp. 23–40. 2
- [WFH\*01] WARE M., FRANK E., HOLMES G., HALL M., WITTEN I. H.: Interactive machine learning: Letting users build classifiers. *International Journal of Human-Computer Studies* 55, 3 (2001), 281–292. doi:10.1006/ijhc.2001.0499. 3
- [WOBM17] WYNER A. J., OLSON M., BLEICH J., MEASE D.: Explaining the success of adaboost and random forests as interpolating classifiers. *Journal of Machine Learning Research* 18, 1 (Jan. 2017), 1558–1590. 2
- [Wol92] WOLPERT D. H.: Stacked generalization. *Neural Networks* 5, 2 (1992), 241–259. doi:10.1016/S0893-6080(05)80023-1. 2
- [WZWY21] WANG J., ZHANG W., WANG L., YANG H.: Investigating the evolution of tree boosting models with visual analytics. In *2021 IEEE 14th Pacific Visualization Symposium (PacificVis)* (2021), pp. 186–195. doi:10.1109/PacificVis52677.2021.00032. 3
- [XCC\*21] XIA Y., CHENG K., CHENG Z., RAO Y., PU J.: GBMVis: Visual analytics for interpreting gradient boosting machine. In *Cooperative Design, Visualization, and Engineering* (Cham, 2021), Springer International Publishing, pp. 63–72. 3
- [YNB21] YUAN J., NOV O., BERTINI E.: An exploration and validation of visual factors in understanding classification rule sets, 2021. arXiv: 2109.09160. 3
- [ZC18] ZHOU J., CHEN F.: 2D transparency space—Bring domain users and machine learning experts together. In *Human and Machine Learning*. Springer, 2018, pp. 3–19. doi:10.1007/978-3-319-90403-0\_1. 2
- [Zho09] ZHOU Z.-H.: *Ensemble Learning*. Springer US, Boston, MA, 2009, pp. 270–273. doi:10.1007/978-0-387-73003-5\_293. 2
- [ZHPA13] ZOU J. Y., HSU D. J., PARKES D. C., ADAMS R. P.: Contrastive learning using spectral methods. In *Advances in Neural Information Processing Systems* (2013), vol. 26, Curran Associates, Inc. 3, 7
- [ZWLC19] ZHAO X., WU Y., LEE D. L., CUI W.: iForest: Interpreting random forests via visual analytics. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (Jan. 2019), 407–416. doi:10.1109/TVCG.2018.2864475. 2, 3, 6, 8