# Interactive Exploration of Large Decision Tree Ensembles

**Jan Forberg**
jan.forberg@mailbox.tu-dresden.de
Interactive Media Lab Dresden
Technische Universität Dresden
Dresden, Germany

**Annett Mitschick**
annett.mitschick@tu-dresden.de
Interactive Media Lab Dresden
Technische Universität Dresden
Dresden, Germany

**Martin Voigt**
martin.voigt@ai4bd.com
AI4BD Deutschland GmbH
Dresden, Germany

**Raimund Dachselt**
dachselt@acm.org
Interactive Media Lab Dresden
Technische Universität Dresden
Dresden, Germany

## ABSTRACT

In this paper, we present a visual exploration tool for tree-ensemble based prediction models in order to make them more interpretable and explicable for users with little machine learning knowledge. Tree ensembles like Gradient Boosted Trees are commonly used for prediction tasks. While their performance is outstanding, their lack of interpretability still limits their broader use in the business world. We propose a novel approach for presenting specific features distributed throughout large tree sets to render decisions intelligible. In a preliminary user study with machine learning consultants we collected valuable feedback which strongly hints towards the usefulness of our approach.



*CHI '19 HCMLP Workshop, May 04, 2019, Glasgow, Scotland, UK*



ACM ISBN 978-1-4503-5971-9/19/05…$15.00
https://doi.org/10.1145/1122445.1122456

## KEYWORDS

predictive models, tree ensembles, visual exploration, interpretability

## INTRODUCTION

In the past few years predictive models such as tree ensembles have been adapted by professionals of different domains to leverage prediction processes. However, machine learning laymen still struggle with the task of truly understanding the model behavior and to trust the predictions. For sensible decisions relying on such potentially untrustworthy gray-box prediction models is risky. By developing a simple, interactive exploration tool for tree-ensemble-based prediction models we aim to improve their explicability and interpretability even for users with little to no machine learning (ML) knowledge.

## RELATED WORK

A wide range of research throughout the past two decades is targeted towards explicability and simplification of prediction models. Most proposed methods are aimed towards the creation of simplified surrogate models and data analysis [2, 3, 5, 9, 11]. One of the most popular frameworks for classifier analysis is *LIME* (Local Interpretable Model-Agnostic Explanations) by Ribeiro et al. [8], especially for computer vision tasks.

Only few propose solutions based on interactive visualization. Lakkaraju et al. [4] propose interpretable decision sets. As opposed to decision lists, each element (i. e., a rule) of a decision set applies independently and is shown to be more interpretable to humans than the widely used if-then-else structured lists. Van den Elzen & van Wijk [10] have presented BaobabView, an application that can be used for both decision tree construction and data analysis for single-tree classifiers. *BigML*[1] is a lay user friendly, yet commercial, platform for machine learning related visualization, rendering common model statistics, such as training data histograms, PDPs and single tree visualizations (common hierarchies, sunburst). An approach similar to the concepts presented in this work is *RuleMatrix* by Ming et al. [7] which has been released recently. Prior to visualization, an if-then-else structured rule list is extracted from the examined model. Rules can then be filtered and displayed in a matrix.

[1] https://bigml.com

## A COMPREHENSIBLE VISUALIZATION OF TREE ENSEMBLES BASED ON RULE SETS

While the visualization of a single tree is usually easy to interpret (at least if the features are interpretable), the visualization of a large forest consisting of more than 100 trees is much more problematic. As decisions involving a specific feature are distributed throughout the ensemble, browsing numerous sets of single-tree visualizations does not add to the comprehensibility. Therefore, we created a rule-based model which enables an easier examination of specific model features.

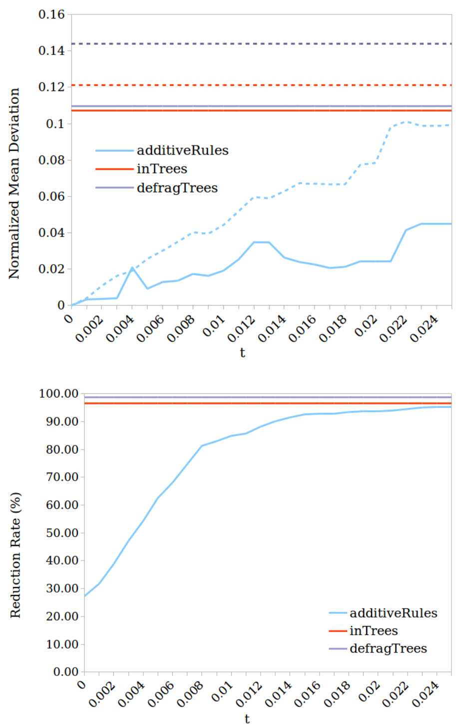


**Figure 1: Normalized mean deviation (top) and reduction rate (below) of the presented approach (*additiveRules*) compared to the rule extraction methods *inTrees* [1] and *defragTrees* [3] for a sample data set. $t$ is a threshold parameter for impact and entropy pruning. The dotted line denotes the deviation for randomly generated data, the continuous line denotes the deviation for the training data points. The reduction rate is based on the number of rules emitted by an unpruned mapping from forest to rule set.**

[2] https://www.kaggle.com/dongeorge/beer-consumption-sao-paulo

## Rule Extraction

A decision tree can easily be converted to a set of rules by simply interpreting each distinct path from root to leaf as a rule: Each split node denotes a rule condition, whereas the rule value is set to the leaf value. Additionally, simplification methods can be applied which reduce the amount and complexity of the rule set while introducing only minor deviations from the predictions of the original model. The following pruning methods have been applied:

- *Entropy Pruning:* During rule extraction, stop the traversal from root to leaf once a node with an entropy below a certain threshold $T_{entropy}$ has been found. The value of the emitted rule is set to an average of the leaf values below this node.
- *Impact Pruning:* After rule extraction, remove rules with values below a certain threshold $T_{impact}$.
- Reduce rules to one condition per feature. Merge conditions targeting the same feature into a single condition. This step reduces rule complexity without impacting the prediction result.
- Merge two *complementary* rules into a single rule and a base prediction value. Simplify rules only differing in a single complementary condition.

By adjusting the thresholds, relatively small rule sets (less than 100 rules) could be created from forests consisting of around 100-150 trees while sustaining reasonable precision. Some evaluation results for a sample data set[2] are shown in Figure 1.

## Visualization and Interaction

Throughout our incremental development process, we received feedback from machine learning consultants and clients (machine learning laymen). This process led us to a visualization tool for model analysis with two views as described in the following.

*Interactive Rule View.* A set of rules is represented by vertically stacked groups of sliders (see Figure 2). Each slider represents a rule condition with a colored interval on its range. A rule may consist of multiple sliders, depending on the number of conditions. While the slider enables the user to change a feature value interactively, the colored range denotes the subset of all possible values for which the rule is evaluated as “true”. If all conditions are met the rule is highlighted in light green. A number of sorting and filtering tools were added to enable the exploration of the rule set. Rules can be sorted by any feature, value, size (i. e., number of conditions) and state (i. e., *true* or *false*). Each feature of the model can be locked and/or hidden. Hiding a feature simple hides all associated condition sliders. Locking can be useful for features that cannot be influenced (e. g., the current month). Activating the locked state of a feature disables all associated sliders and hides all rules that cannot evaluated as “true” with the locked feature value.

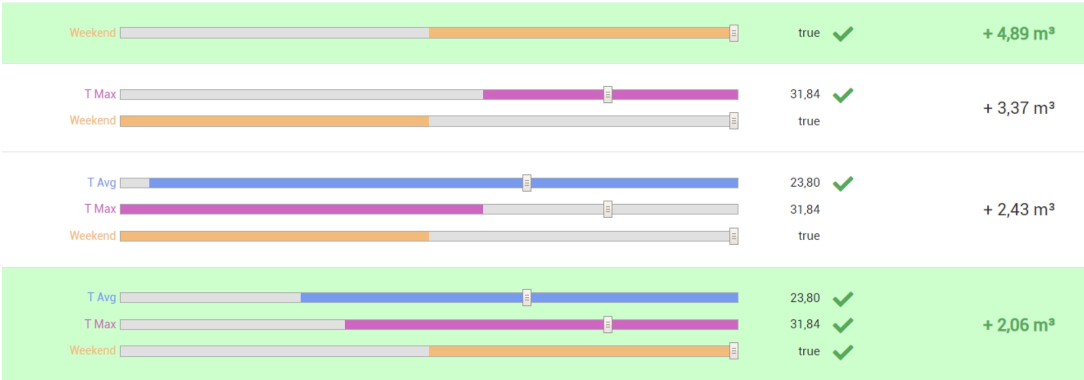


**Figure 2: Vertically stacked rule views. Each rule consists of a set of condition sliders combined with a value on the right hand side. Rules evaluated as "true" are highlighted in green.**

*Interactive Feature View.* While the described rule view was considered helpful when trying to understand the model and led users to the question: "Where do I go from here?". While the global influence of a single feature on the prediction result is usually very complex and dependent on other features, the local influence can be calculated by simply evaluating the model in the local neighborhood. The application was therefore extended by a second view which introduces the concept of *Dynamic Local Dependence Plots* view (see Figure 3). For each model feature and predictive function, a Dynamic Local Dependence Plot is an adaptive bar graph showing the relation between feature values and prediction label for an interactively changeable input. The subdivision of feature value range into non-overlapping segments both facilitates the visualization and limits the problem of dynamic reevaluation of the segment values. The height of each rectangle is scaled depending on the simulated prediction. As the presented plots show the local model behavior, they give no indication of global trends. Changing input values results in a visual transformation of all plots which indicates a change of the local behaviors.

While Dynamic Local Dependence Plots can help the user to learn about the model behavior within certain constraints, this process can be automated. The interactive application was extended to include means to set simple constraints for each feature (i. e. to restrict the range of a feature) and find the highest or lowest possible model prediction within these bounds. To our knowledge, this problem can only by tackled by evaluating the model for every possible feature configuration (within the applied constraints). As this can take several hours for complex models with a large amount of distinct splits, the global maximum is only approximated to provide the result immediately.

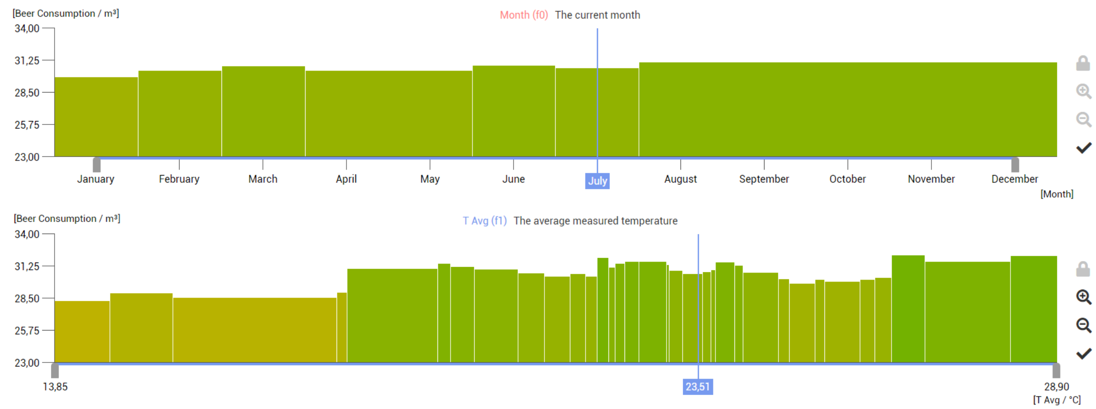


**Figure 3: A screenshot of the feature view showing two features of the example data set: month and average temperature.**

## EVALUATION

We evaluated the preliminary prototype in a think-aloud study with five machine learning consultants (age between 26 and 34). Each participant was asked to perform some exploration tasks with the visualization tool in a session of around 40-50 minutes. The study showed that most application functions could be learned easily. Especially the feature view with the Dynamic Local Dependence Plots were well-received. Additionally, the rule visualization was well-interpretable by study participants. In general, the participants considered the feature view to be very helpful for machine learning laymen due to its simplicity. As suggested by one of the participants, a top-down navigation from feature view (general overview) to rule view (detailed search) could constitute a reasonable extension of this work.

At the end of each session, the participants were asked to fill out a PSSUQ (Post Study System Usability Questionnaire) [6]. Due to the limited amount of testers, this questionnaire has to be interpreted carefully, but indicates a rough trend of how different aspects of the prototype were received. The best average scores in the PSSUQ were given for the statements “I believe I could become productive quickly using this system” (6,4), “The interface of this system was pleasant” (6,4) and “Overall, I am satisfied with this system” (6,2).

## CONCLUSION AND FUTURE WORK

The developed visualization tool is geared towards machine learning laymen in order to make tree-based predictive models more comprehensible by displaying an accurate substitute model in form of rule sets. Both presented views (rule list and Dynamic Local Dependence Plots) show the model "as is" with only minor simplifications applied during rule extraction. Feedback from users has been taken into consideration during development, however, a user study with machine learning laymen has not been conducted in the scope of this work. Preliminary results strongly hint towards the usefulness and comprehensibility of the visualizations for users without prior ML knowledge as well as its applicability for explanatory tasks, e. g., when explaining a prediction model to a client.

A range of extensions has been suggested during the user study. A more central role of the feature view has been proposed including means of navigation to the rule view. The visual distinction of realistic and unrealistic data points is another considerable suggestion. While the closely related rule coverage has been ignored during rule extraction, its visualization can be beneficial when examining unusual peaks in the Local Dependence plots. Since the current implementation still involves a lot of scrolling, an exploration of the concepts on large displays would be interesting. The Dynamic Local Dependence plots could be extended by Partial Dependence Plots. This could facilitate the assessment of unusual local effects while providing a clear distinction between local and global behavior. Furthermore, the actual explanatory nature of visualized additive rule sets has to be assessed in another study involving machine learning laymen and a bigger group of participants.

## REFERENCES

[1] Houtao Deng. 2018. Interpreting tree ensembles with inTrees. *Int. Journal of Data Science and Analytics* (2018), 1–11.

[2] Jerome H. Friedman and Bogdan E. Popescu. 2008. Predictive learning via rule ensembles. *Ann. Appl. Stat.* 2, 3 (09 2008), 916–954. https://doi.org/10.1214/07-AOAS148

[3] Satoshi Hara and Kohei Hayashi. 2018. Making Tree Ensembles Interpretable: A Bayesian Model Selection Approach. In *Proc. of the 21st Int. Conf. on Artificial Intelligence and Statistics*. 77–85. http://proceedings.mlr.press/v84/hara18a.html

[4] Himabindu Lakkaraju, Stephen H. Bach, and Jure Leskovec. 2016. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. ACM, 1675–1684.

[5] Benjamin Letham, Cynthia Rudin, Tyler H. McCormick, and David Madigan. 2013. An Interpretable Stroke Prediction Model using Rules and Bayesian Analysis.. In *AAAI (Late-Breaking Developments)*.

[6] James R. Lewis. 1992. Psychometric evaluation of the post-study system usability questionnaire: The PSSUQ. In *Proc. of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 36. SAGE Publications Sage CA: Los Angeles, CA, 1259–1260.

[7] Yao Ming, Huamin Qu, and Enrico Bertini. 2018. RuleMatrix: Visualizing and Understanding Classifiers with Rules. *IEEE transactions on visualization and computer graphics* (2018).

[8] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. 1135–1144.

[9] Naphaporn Sirikulviriya and Sukree Sinthupinyo. 2011. Integration of rules from a random forest. In *International Conference on Information and Electronics Engineering*, Vol. 6. 194–198.

[10] Stef Van Den Elzen and Jarke J van Wijk. 2011. Baobabview: Interactive construction and analysis of decision trees. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*. IEEE, 151–160.

[11] Fulton Wang and Cynthia Rudin. 2015. Falling rule lists. In *Artificial Intelligence and Statistics*. 1013–1022.

**ACKNOWLEDGEMENTS**

Funded by the Deutsche Forschungsgemeinschaft (DFG) – TRR 248, project number 389792660 (see https://perspicuous-computing.science).