Explaining Random Forest Predictions with Association Rules

Henrik Boström, Ram B. Gurung, Tony Lindgren and Ulf Johansson

Abstract Random forests frequently achieve state-of-the-art predictive performance. However, the logic behind their predictions cannot be easily understood, since they are the result of averaging often hundreds or thousands of, possibly conflicting, individual predictions. Instead of presenting all the individual predictions, an alternative is proposed, by which the predictions are explained using association rules generated from itemsets representing paths in the trees of the forest. An empirical investigation is presented, in which alternative ways of generating the association rules are compared with respect to explainability, as measured by the fraction of predictions for which there is no applicable rule

Henrik Boström
KTH Royal Institute of Technology, School of Electrical Engineering and Computer Science,
Electrum 229, 164 40 Kista, Sweden
✉ `bostromh@kth.se`

Ram B. Gurung
Dept. of Computer and System Sciences, Stockholm University, Sweden
✉ `gurung@dsv.su.se`

Tony Lindgren
Dept. of Computer and System Sciences, Stockholm University, Sweden
✉ `tony@dsv.su.se`

Ulf Johansson
Dept. of Computer Science and Informatics, Jönköping University, Sweden
✉ `ulf.johansson@ju.se`

Archives of Data Science, Series A
(Online First)
KIT Scientific Publishing
Vol. 5, No. 1, 2018

DOI: 10.5445/KSP/1000087327/05
ISSN 2363-9881

and by the fraction of predictions for which there is at least one applicable rule that conflicts with the forest prediction. For the considered datasets, it can be seen that most predictions can be explained by the discovered association rules, which have a high level of agreement with the underlying forest. The results do not single out a clear winner of the considered alternatives in terms of unexplained and disagreement rates, but show that they are associated with substantial differences in computational cost.

1 Introduction

Random forests are frequently shown to achieve state-of-the-art predictive performance (Caruana and Niculescu-Mizil, 2006; Delgado et al, 2014). This performance, however, comes with a cost that is shared with many other high-performing techniques; the logic behind their predictions cannot be easily understood. For a random forest, this is a consequence of forming predictions by averaging, often involving hundreds or thousands of possibly conflicting predictions of individual trees in the forest. In this study, we will investigate means of enabling the understanding the random forest predictions through approaches that aggregate information from predictions of the individual trees. In particular, we consider approaches that represent paths from roots to leaves in the trees as itemsets, which enables the use of frequent itemset mining and association rule discovery techniques to analyse the predictions. The idea of representing forests as itemsets and generating association rules from them was originally proposed in (Deng, 2019). However, in that study, the focus was on providing descriptive summaries of forests and to generate interpretable classifiers using the discovered rules. The idea of explaining predictions of the original forest, which is the focus of this study, was not considered.

The main contributions of the study are:

- The idea of explaining random forest predictions with association rules is proposed.
- Different approaches to generating association rules to explain predictions are suggested.

- An empirical investigation of the explainability of the rules generated by the alternative approaches is presented, where novel performance metrics are suggested and employed.

In the next section, we provide some pointers to, and briefly discuss, related work. In Sect. 3, we describe how association rule mining is proposed to explain random forest predictions, and present different approaches to generating the rules. In Sect. 4, we present results from an empirical investigation comparing the different approaches using two proposed metrics. Finally, in Sect. 5, we discuss the findings and point out directions for future work.

2 Related work

2.1 Association rule mining

The problem of mining association rules was first described by Agrawal et al (1993) for market basket transaction data. Given a database of transactions (or more generally, sets of items, or itemsets), the problem is to find association rules of the form $X \rightarrow Y$, where X and Y are disjoint itemsets. Association rules can be used to predict the occurrence of an item (or a set of items) based on the occurrences of other items in the itemsets. The strength of an association rule is measured in terms of its *support*, i.e., fraction of itemsets for which both X and Y are subsets, and *confidence*, i.e., fraction of the itemsets containing X that also contain Y (Tan et al, 2005). Association rule mining aims to discover all rules in the database that satisfy some minimum levels of support and confidence. Liu et al (1998) first proposed integrating association rule mining with classification, by which first frequent class association rules (CARs), i.e., association rules with a class label as the consequent (Y), are discovered using the Apriori algorithm (Agrawal and Srikant, 1994), from which a subset of rules is selected and ordered according to support. During classification, the first applicable rule in the ordered set is chosen. In order to allow for handling large datasets, more efficient approaches have been proposed, such as CMAR (Li et al, 2001) and MCAR (Thabtah et al, 2005).

2.2 Interpreting opaque models

Rule extraction was introduced for enabling understanding artificial neural networks (ANNs), see Andrews et al (1995). When extracting rules from ANNs, the original approach, called *decompositional* or *open-box*, was to generate rules for individual units within a trained ANN, and then combine these into a rule set. Two well-known open-box algorithms are *RX* (Lu et al, 1995) and *Subset* (Fu, 1991). In *pedagogical* (or *black-box*) rule extraction, the ANN is used to label the training instances, before some standard learning algorithm performs the actual induction of a transparent model, e.g., a rule set, a decision tree or a decision list. Two representative black-box algorithms are *TREPAN* (Craven and Shavlik, 1996) and *G-REX* (Johansson et al, 1997). Since black-box rule extraction algorithms can be applied to any opaque model, including ensembles, most modern rule extractors use that approach, see Huysmans et al (2006).

The need for getting some insights into random forests was already addressed in the original work by Breiman (2001), in which an approach to calculating the *variable importance* was proposed, which measures the effect on predictive performance when permuting the values for each variable. This approach has been further developed, see e.g., (Strobl et al, 2008; Henelius et al, 2014). It should be noted that these approaches highlight what features have the highest impact, but do not explain how they affect the predictions.

The *inTree* (interpretable tree) framework proposed in (Deng, 2019) provides an interpretable view of tree ensembles, such as random forests, by employing association rule mining to itemsets generated from each path from a root to a leaf in the forest, where each condition on the path and the prediction in the leaf becomes an item in the itemset. In addition to suggesting that the itemsets are summarized by association rules, with specified levels of support and confidence, an approach to generating an interpretable classifier by selecting a subset of the rules was presented. The latter can be seen as a form of rule extraction, which however contrasts to previous approaches by the use of the original dataset when generating the classifier, rather than a dataset labeled by the opaque model. Hence, this approach aims for high accuracy rather than high fidelity to the underlying forest.

Another approach, addressing a similar problem, was proposed by Friedman et al (2008), by which sets of rules are extracted from the trees in a forest that are used as features in a linear model, for which interpretability is enhanced by employing the Lasso. Again, the approach is hence not used for explaining predictions of the original forest, but rather to provide an interpretable approximation of it.

2.3 Explaining predictions

One of the more well-known approaches to explain predictions of opaque models is LIME (Local Interpretable Model-agnostic Explanations) (Ribeiro et al, 2016). It employs a mapping from the original feature space into an interpretable feature space, and explains a prediction in this space by creating an interpretable model from instances that are sampled in the region around the instance to be predicted, using the underlying opaque model to label the instances. DALEX (Biecek, 2018) utilizes a similar approach to construct explanations for a certain prediction, where the model is complemented with a wide variety of plots to further enhance interpretability of the generated local model. In (Lundberg and Lee, 2017), a unified framework for interpreting model predictions is presented, capturing LIME as well as additional approaches. However, it should be noted that the explanations provided by these approaches agree only locally with the opaque model, i.e., in some specific region surrounding the test instance, while the explanations provided by the approaches considered in this study are not conditioned on any such (implicit) region, other than what is explicitly stated in the discovered rules.

3 Methods

In this section, we first describe four approaches to generating itemsets from random forests that are considered in this study. We then describe how the itemsets are to be analysed using association rule mining. Finally, we describe how the discovered rules are used to explain predictions of new test instances.

3.1 Approaches to generating itemsets

All considered approaches for generating itemsets are based on the original idea proposed in (Deng, 2019), by which each path from a root of a tree in a forest to a leaf is represented by an itemset, where the items correspond to the conditions, i.e., triples consisting of a feature (variable), operator and value, encountered on the path, together with the prediction (class label) of the leaf node. See Figure 1, for an example of a tree and the corresponding itemsets that are generated from it.

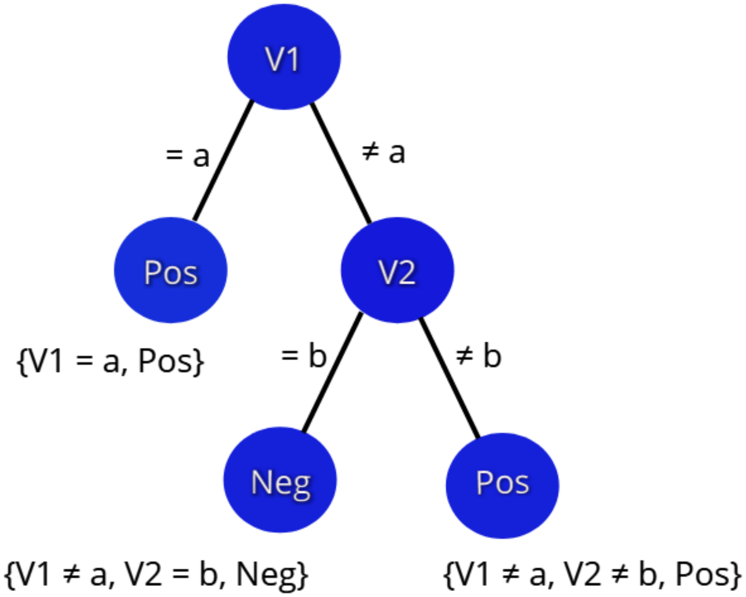


Figure 1: Representing paths as itemsets.

In this study, we consider binary classification trees with categorical features only with no missing values. Hence, numerical features have to be discretized (binned), and, as a consequence, the operators of the conditions are limited to equality (=) and inequality (≠). Moreover, following the recommendation by Kliegr et al (2018) that “artificial learning systems should refrain, wherever feasible, from the use of negation in the discovered rules that are to be presented to the user", we have here chosen to eliminate conditions involving inequality from the generated itemsets. In addition to increased interpretability, this operation also leads to the consequence that the average size of the itemsets are approximately halved, which has a positive effect on the computational cost when finding rules. However, this may on the contrary have a negative effect on the possibility of actually finding any rules with high confidence.

The four considered alternative approaches to generating itemsets are:

Alternative 1: One itemset per leaf. This approach is the same as used in the *inTree* system (Deng, 2019), i.e., one itemset is generated for each leaf in the forest. The resulting collection of itemsets is represented by a bag, i.e., there may be duplicate itemsets in case the same path appears in multiple trees in the forest. The size (s) of the bag of itemsets generated by this approach is $s = t \cdot l$, where t is the number of trees in the forest and l is the average number of leafs in the trees. The average size of the itemsets is $a = d + 1$, where d is the average length of the paths from the roots to the leafs in the forest (the itemsets contain d conditions on average plus one class label).

It should be noted that the itemsets generated by this approach do not reflect the frequencies of training instances falling into the leafs of the trees. This means that a particular subset of conditions that are shared by a relatively large proportion of the training instances may be represented by fewer itemsets than another subset of conditions, shared by fewer training instances, simply because there are more paths including the latter subset. Taken to the extreme, a vast majority of the training instances may be represented by a single itemset, while a small minority of the training instances may be represented by a very large number of itemsets. This means that support and confidence calculated from itemsets generated by this approach are not directly connected to coverage, in terms of training instances, and hence may reflect the underlying class probability distribution poorly when explaining predictions using these rules.

Alternative 2: One itemset per tree prediction. The second approach aims for remedying the potential weakness of not encoding frequencies by the first approach, simply by making a prediction for each training instance, using the random forest generated from the same set of instances. Again, each path from a root in a tree to a leaf will correspond to an itemset. Since each leaf in the forest will include at least one training instance, the set of itemsets considered by this approach will be the same as the set of itemsets generated by the first approach. However, as we consider multi-sets (bags) of itemsets, there will be a significant difference; a leaf will by the second approach be represented by the same number of (identical) itemsets as the number of training instances that fall into that leaf, hence encoding the frequencies by duplicating itemsets. It should be noted that the second approach hence may lead to substantially larger bags, which may incur a large additional computational cost during association rule discovery. Only in the special case, where each leaf corresponds to exactly one training instance will the bags be of the same size. This special case, however, does not occur in practice, i.e., *almost surely*, as each tree is generated from a bag of the training instances, leading to that the out-of-bag instances will fall into already occupied leafs, when predictions are made for these. The size of the bag of itemsets generated by this approach is hence $s = t \cdot n$, where n is the number of training instances. The average size of the itemsets is the same as for the first approach, i.e., $a = d + 1$.

Alternative 3: One itemset per training instance. It should be noted that the two first approaches only consider predictions made by the individual trees, and in particular do not consider the forest predictions, i.e., the averaged vote of the trees involved in a prediction. Since we are aiming for explaining predictions of forests, rather than individual trees, a natural alternative to the above approach is to consider the union of itemsets of the leafs involved in a prediction, but where all class labels obtained from the individual leafs are replaced by the single class label with the highest predicted probability according to the forest. By this approach, there will be only one itemset per training instance, i.e., $s = n$. The average size of the itemsets will however increase, and an upper bound of this size is $a \leq t \cdot d + 1$. However, the average itemset size will typically be smaller than the bound, since different paths may share conditions.

Alternative 4: One itemset per training instance, excluding disagreement. Many forest predictions are formed from individual tree predictions where possibly a substantial part of the trees are not in agreement with the majority vote. This is the case in particular when dealing with multiclass problems, where often only a minority is supporting the forest prediction. Including itemsets corresponding to paths from such disagreeing predictions can be expected to make the task of finding high-confidence association rules more difficult. The fourth, and final, proposed strategy is hence to modify the previous approach by forming itemsets by taking the union of only the individual itemsets for which the class label is the same as the forest prediction. Similarly to the previous approach, there will be only one itemset per training instance, i.e., $s = n$, and the bound of the average size of the itemsets will be the same, i.e., $a \leq t \cdot d + 1$. However, the actual average is reduced compared to the previous, when there is a low level of agreement and an overlap of conditions among the individual trees.

3.2 Association rule mining and filtering

From the itemsets generated by the above approaches, association rules are searched for with specified levels of *confidence* and *support*. The reason for focusing on confidence rather than other notions of *interestingness* (Bayardo and Agrawal, 1999; Geng and Hamilton, 2006) is that we want to find rules that are in agreement with the forest predictions, rather than e.g., have a high lift with respect to one of the class labels. The association rules are here restricted to have a (single) class label in the consequent, and to have a set of conditions (variable, operator and value triples) in the antecedents.

Similar to previous approaches for generating classifiers from association rule minining, such as CBA (Liu et al, 1998), a filtering step is employed to produce a small set of rules with as high confidence as possible. More precisely, a rule is kept if there for some training instance is no other applicable rule with higher confidence or with the same confidence and higher support.

3.3 Explaining predictions

The above approaches for generating itemsets can be directly applied also to (test) instances that have not been included in the training set. For a single test instance, the two first approaches will result in a set of itemsets, corresponding to all paths from a root to a leaf into which the test instance falls. For the third and fourth approach, only a single itemset will be generated, which is obtained by merging the itemsets generated from the individual trees, but with the individual class labels replaced by the predicted class label. In addition, the fourth approach includes only itemsets from trees that agree with the forest prediction.

Each previously discovered association rule may then be applied to the test instance, as represented by one or more itemsets. We say that a rule is *applicable* with respect to a test instance, if and only if, the antecedents of the rule is a subset of an itemset of the test instance. It can further be checked whether the consequent of an applicable association rule agrees with the forest prediction of an instance. If that is the case, we say that the rule can *explain* the forest prediction. It should be noted that for some test instances there may be no applicable rule, and such instances are referred to as *unexplained*. In the extreme case, no association rules with specified levels of support and confidence can be found, resulting in the fact that no predictions can be explained. Note also that for some test instances, there may be multiple applicable rules. If more than one rule have the same consequent as the forest prediction, they may be considered as alternative explanations. Any applicable rule having a class label in the consequent that differs from the forest prediction is considered to be *disagreeing*.

Clearly, the levels of support and confidence, and the approach for generating itemsets, may have an impact on the extent to which (test) predictions can be explained as well as the level of (dis)agreement among applicable rules. In the next section, we will empirically investigate these effects on some standard datasets.

4 Empirical investigation

4.1 Experimental setup

Eight classification datasets from the UCI Machine Learning Repository (Dheeru and Karra Taniskidou, 2017) were used in the experiment: Car evaluation, Ecoli, Glass, Iris, Thyroid, Pima Indians diabetes, Tic-tac-toe endgame and Wine. The number of classes ranges from two to seven, the number of attributes from 4 to 13 and the number of instances from 150 to 1728 instances. Real-valued features were binned into 10 categories of equal size.

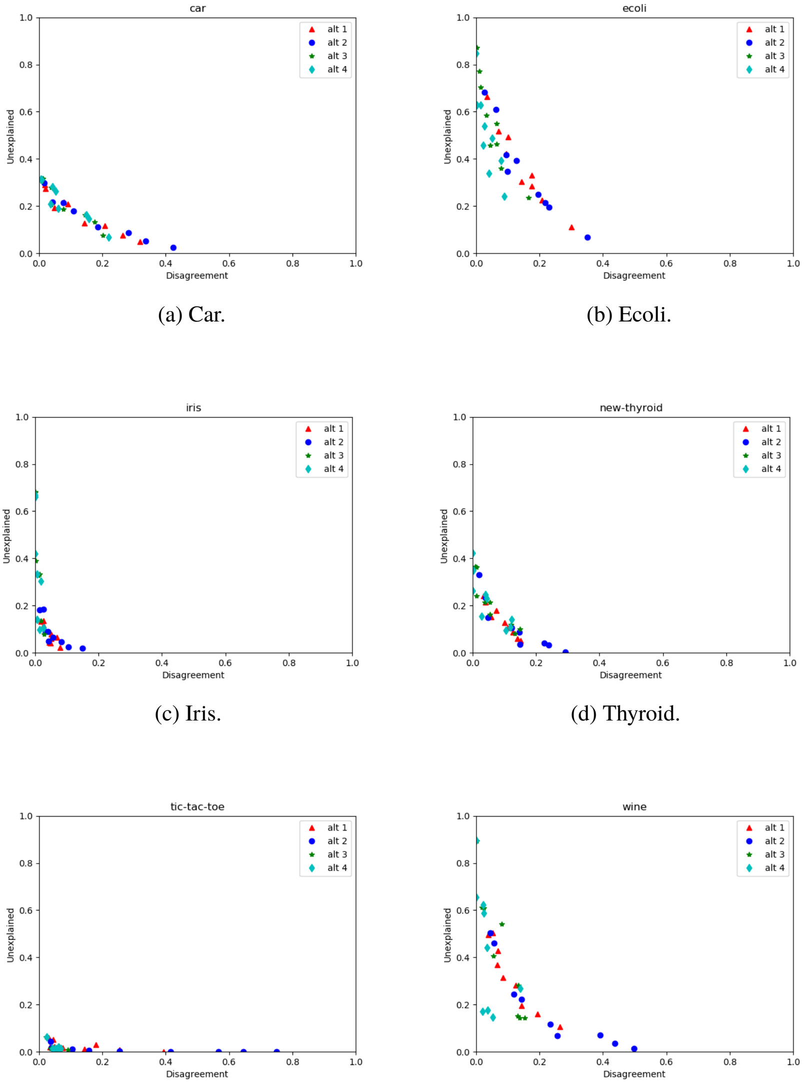
The following two performance metrics were used: i) fraction of test instances for which there are no applicable rules (*unexplained*), and ii) fraction of test instances for which there is at least one applicable rule with a different consequent (class label) than the forest prediction (*disagreement*). Each dataset was randomly split into 50% training instances and 50% test instances and random forests with 100 trees were generated from the training instances. The reported results are averages of ten repetitions.

The same set of confidence values were considered for all four approaches; $\{0.8, 0.9, 1.0\}$. However, as the number of itemsets produced by the approaches differ (except for the third and fourth approach), feasible support values differ between the approaches, e.g., using a support of less than 0.5% is not meaningful for the two latter approaches as the rules would be allowed to cover single training instances for some of the datasets, while this is still a too high value to allow any rules to be found by the two first approaches, i.e., these would leave all test instances unexplained. In order to obtain a wide spread in the unexplained and disagreement rates for all approaches, the following support values (in absolute numbers) were considered: $\{10, 15, 20\}$ for alternative 1, $\{25, 50, 75\}$ for alternative 2, and $\{5, 10, 15\}$ for alternative 3 and 4.

The random forest algorithm together with the above approaches to explain predictions were implemented in Python 3.6.6, using the following libraries: Pandas 0.23.4, Numpy 1.15.1, sharedmem 0.3.5 and mlxtend 0.13.0, and the experiments were executed on a HP Zbook 15 with an Intel Xeon CPU E3-1505M v5 (2.80GHz, 4 physical cores) and 32 GB PM, using Ubuntu 18.04. A re-implementation of the random forest algorithm was preferred to the standard Scikit-learn implementation (Pedregosa et al, 2011), to simplify handling of categorical features and extraction of paths from predictions. Although sharing some underlying ideas with the R package *inTrees* (https://CRAN.R-project.org/package=inTrees), the two implementations are completely independent. The source code is available upon request from the first author.

4.2 Experimental results

4.2.1 Empirical findings



(a) Car.

(b) Ecoli.

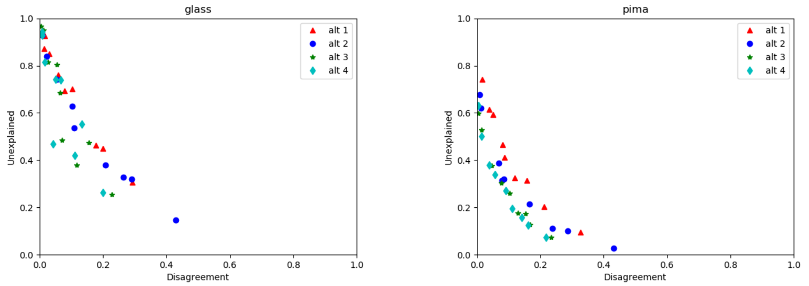
(c) Iris.

(d) Thyroid.

(e) Tic-tac-toe.

(f) Wine.

Figure 2: Unexplained and disagreement rates.



(g) Glass.

(h) Pima.

Figure 2: Unexplained and disagreement rates.

In Figure2, the unexplained and disagreement rates are shown for the three alternative approaches, using the various parameter settings for support and confidence, for the eight datasets. One can see that the investigated parameter values allow for various trade-offs between the unexplained and disagreement rates. There is no clear winner in the sense that one method dominates the others, i.e., appears on the Pareto front on all datasets; although for some datasets the third and fourth approach outperform the others for almost all parameter settings. For disagreement rates of up to around 10%, less than 20% of the instances are left without explanation for six of the datasets (for Ecoli, a slightly higher unexplained rate is observed, while for Glass, it is substantially higher).

Table 1: Averages over all parameter settings and results.

Alt.	Itemsets	Itemset size	Freq. itemsets	Unfiltered	Filtered	Unexplained	Disagreed	Time
1	5559.57	3.49	1111.15	42.91	27.35	0.27	0.11	84.07
2	28412.50	3.10	970.85	66.88	39.62	0.22	0.18	220.62
3	284.13	8.48	2477.51	355.83	23.21	0.34	0.07	4.93
4	284.13	8.40	2475.03	356.99	23.12	0.34	0.05	4.99

The perhaps most noticeable difference between the approaches is the number of itemsets that are generated and the associated computational cost to process them. In Table 1, the average results over all parameter settings and datasets are shown, where the first three columns, following the one indicating the alternative, show the average number of itemsets, the average number of items in the itemsets and the average number of frequent itemsets. The following two columns (unfiltered and filtered) refer to the number of discovered association rules before and after filtering, respectively. The last column (Time) shows the average total computation time in seconds, involving all model building and rule generation steps as well as the application of rules to explain the predictions. It should be noted that the reported times are affected by the actual implementation and the (lack of) considered optimizations, so any comparison based on the reported times should be made with some caution. However, as the computational cost is directly related to the number of itemsets, and their average sizes, the reported times can be seen as a confirmation of the cost associated with these numbers.

4.2.2 Example rules

To illustrate the result of explaining random forest predictions with association rules, we present some example rules, generated by alternative 4 with a minimum support of five instances and confidence of 1.0, on two of the datasets, Tic-tac-toe and Pima Indians Diabetes, for which a random half is used for training and the other half for testing, similar to the above experiment.

Tic-tac-toe is a well known board game where each player in turn place their symbols, either 'o' or 'x', on the board, i.e., a 3 x 3 grid, with the aim of getting three symbols in a row, horizontally, vertically or diagonally. Table 2 shows the top rules for the two classes; 'win for x' (positive) and 'no win for x' (negative). In addition to the antecedents and consequents, also the number of training instances for which the corresponding rule is the first applicable is shown (Count).

Table 2: Association rules for Tic-tac-toe.

Antecedents	Consequent	Count
top-left-sq. = x, middle-middle-sq. = x, bottom-right-sq. = x	positive	50
top-right-sq. = x, bottom-left-sq. = x, middle-middle-sq. = x	positive	45
top-right-sq. = x, top-left-sq. = x, top-middle-sq. = x	positive	42
...		
top-left-sq. = o, middle-middle-sq. = o, bottom-right-sq. = o	negative	24
top-left-sq. = o, top-right-sq. = o, top-middle-sq. = o	negative	23
top-middle-sq. = o, bottom-middle-sq. = o, middle-middle-sq. = o	negative	21
...		

It should be noted that capturing the true target function using a single decision tree would require hundreds of leaf nodes, due to the *replication problem* (Bagallo and Haussler, 1990), i.e., the difficulty of expressing disjunctive concepts using trees. In contrast, all instances of the positive class can be defined using only eight association rules, which are easily discovered from itemsets generated by any of the approaches considered in this study. Similarly, the eight cases where three 'o' appear in a row are also found. However, since the negative class also contains instances where none of the symbols appear in a row, providing a complete definition is challenging also when using association rules. This is illustrated by some incorrect rules (predicting the negative class) occasionally being "discovered" in addition to the 16 correct ones. For some test instances, there are multiple applicable rules that all agree with the (correct) forest predictions. It is also worth noting that in this domain, for which the shortcomings of single decision trees is not completely remedied by using forests, which here contain more than 9 000 leaf nodes, the discovered rules occasionally (correctly) disagree with the forest prediction.

The second dataset used to illustrate the explanation facility is Pima Indians Diabetes, which includes the following attributes: number of times pregnant, plasma glucose concentration after 2 hours in an oral glucose tolerance test (gtt), diastolic blood pressure (mm Hg) (dbp), triceps skin fold thickness (mm) (skin-thick), 2-Hour serum insulin (mu U/ml) (2-hour-ins), body mass index (bmi), diabetes pedigree function (dpf) and age. The instances are labeled 'no diabetes' and 'diabetes'. In Table 3, selected association rules are presented for both classes. Note that the numerical features have been discretized and hence the categorical feature values correspond to intervals.

Table 3: Association rules for Pima Indians Diabetes.

Antecedents	Consequent	Count
dbp = (61.0, 73.2], gtt = (159.2, 179.1]	diabetes	8
gtt = (139.3, 159.2], 2-hour-ins = (-0.846, 84.6], dpf = (0.312, 0.546]	diabetes	7
age = (39.0, 45.0], 2-hour-ins = (84.6, 169.2]	diabetes	6
...		
bmi = (20.13, 26.84], age = (20.94, 27.0]	no diabetes	48
bmi = (20.13, 26.84], skin-thick = (9.9, 19.8]	no diabetes	7
dpf = (0.0757, 0.312], gtt = (79.6, 99.5]	no diabetes	23
...		

Examining the first association rule in Table 3, the first condition refers to blood pressure, which here is in the normal range according to National Heart Foundation of Australia (2016), while the second condition concerns the outcome of a glucose tolerance test, see (Carrillo, 2013) for details, for which the considered values are far above what is typically considered normal (less than 140 mg/dL). Hence, the first rule seems to provide a relevant indication of diabetes. The second association rule also utilizes the second attribute, again with values that are higher than what is considered to be normal. Also the second and third conditions of this rule are consistent with known indications of diabetes, see (Saxena P, 2011) and (Smith et al, 1988), respectively. The first condition of the third association rule, stating that the age is between 39 and 45 is almost inline with the known risk factor of being 45 years of age or older (United States Department of Health and Human Services (HHS), 2016), while the second condition again is indicative of diabetes. The remaining three rules define subgroups of the population with low risk of diabetes, again consistent with known risk factors (United States Department of Health and Human Services (HHS), 2016). Again, this example shows that the predictions of a practically opaque random forest, containing more than 10 000 leaf nodes, can be explained using a set of fairly easily interpretable rules, providing the user the possibility to reason about a prediction, possibly question or confirm it, and in the best case, also to gain novel insights.

5 Concluding remarks

We have investigated the use of association rule mining to explain random forest predictions. Four different approaches to representing the predictions using itemsets have been proposed and results from an empirical investigation have been presented, measuring the extent to which test instances can be explained by the discovered rules. The performance of the four approaches differs in terms of the unexplained and disagreement rates, but not consistently over the considered datasets. However, there is a substantial difference in computational cost, where the approaches representing each instance with a single itemset, i.e., alternative 3 and 4, are one to two orders of magnitude faster than the other approaches.

There are a number of possible directions for future work. One concerns the generation of itemsets without requiring the random forest to be trained on discretized numerical features, as discretization may have a negative effect on predictive performance and therefore normally should be avoided. How to effectively handle missing values is another open question. Another direction for future work is to investigate additional ways of representing the itemsets, in particular more refined ways of combining itemsets obtained from multiple trees, rather than simply merging them. Approaches that aim for optimizing the performance metric of interest, e.g., some specific trade-off between the unexplained and disagreement rates, can also be expected to further enhance performance. More extensive empirical investigations are needed, including assessment of the discovered rules by end-users in different domains, in order to verify that the provided explanations indeed are useful, but also to gather additional requirements on systems for explaining opaque models. Finally, left for future research is also how to extend the investigated approaches to explain predictions for other types of forest, such as regression forests (Breiman, 2001), forests of probability estimation trees (Boström, 2012), forests of survival trees (Hothorn et al, 2004; Ishwaran et al, 2008), quantile regression forests (Meinshausen, 2006) and generalized random forests (Athey et al, 2019). In contrast to forests of classification trees, the contribution of each path is not bounded by a finite set of labels for these other types of forest, and hence alternative approaches are required to explain their predictions.

References

Agrawal R, Srikant R (1994) Fast Algorithms for Mining Association Rules in Large Databases. In: VLDB’94, Proceedings of 20th International Conference on Very Large Data Bases, September 12-15, 1994, Santiago de Chile, Chile, pp. 487–499. ISBN: 15-5860-153-8.

Agrawal R, Imielinski T, Swami A (1993) Mining association rules between sets of items in large databases. In: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, ACM, New York, SIGMOD ’93, pp. 207–216. ISBN: 08-9791-592-5, DOI: 10.1145/170035.170072.

Andrews R, Diederich J, Tickle AB (1995) Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-Based Systems 8(6):373–389. DOI: 10.1016/0950-7051(96)81920-4.

Athey S, Tibshirani J, Wager S, et al (2019) Generalized random forests. The Annals of Statistics 47(2):1148–1178. DOI: 10.1214/18-AOS1709.

Bagallo G, Haussler D (1990) Boolean Feature Discovery in Empirical Learning. Machine Learning 5(1):71–99. ISSN: 0885-6125, DOI: 10.1023/A:1022611825350.

Bayardo RJ Jr., Agrawal R (1999) Mining the most interesting rules. In: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, KDD '99, pp. 145–154. ISBN: 15-8113-143-7, DOI: 10.1145/312129.312219.

Biecek P (2018) DALEX: explainers for complex predictive models. J. Mach. Learn. Res., JMLR.org. 1806.08915.

Boström H (2012) Forests of Probability Estimation Trees. International Journal of Pattern Recognition and Artificial Intelligence 26(2). DOI: 10.1142/S0218001412510019.

Breiman L (2001) Random forests. Machine learning 45(1):5–32. ISSN: 0885-6125, DOI: 10.1023/A:1010933404324.

Carrillo A (2013) Oral Glucose Tolerance Test (OGTT). In: Encyclopedia of Behavioral Medicine, Gellman JR Marc D.and Turner (ed), Gellman JR Marc D.and Turner (ed). Springer, New York, pp. 1389–1389. ISBN: 978-1-441910-04-2, DOI: 10.1007/978-1-4419-1005-9.

Caruana R, Niculescu-Mizil A (2006) An empirical comparison of supervised learning algorithms. In: Machine Learning, Proceedings of the Twenty-Third International Conference, New York, ICML '06, pp. 161–168. ISBN: 15-9593-383-2, DOI: 10.1145/1143844.1143865.

Craven MW, Shavlik JW (1996) Extracting Tree-Structured Representations of Trained Networks. In: Advances in Neural Information Processing Systems, MIT Press, pp. 24–30.

Delgado MF, Cernadas E, Barro S, Amorim DG (2014) Do we need hundreds of classifiers to solve real world classification problems? Journal of Machine Learning Research 15(1):3133–3181. URL: `http://dl.acm.org/citation.cfm?id=2627435.2697065`.

Deng H (2019) Interpreting tree ensembles with intrees. International Journal of Data Science and Analytics 7(4):277–287. DOI: 10.1007/s41060-018-0144-8.

Dheeru D, Karra Taniskidou E (2017) UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences. URL: `http://archive.ics.uci.edu/ml`.

Friedman JH, Popescu BE, et al (2008) Predictive learning via rule ensembles. The Annals of Applied Statistics 2(3):916–954. URL: `https://statweb.stanford.edu/~jhf/ftp/RuleFit.pdf`.

Fu L (1991) Rule Learning by Searching on Adapted Nets. In: Association for the Advancement of Artifical Intelligence, pp. 590–595. URL: `https://www.aaai.org/Papers/AAAI/1991/AAAI91-092.pdf`.

Geng L, Hamilton HJ (2006) Interestingness measures for Data Mining: A survey. ACM Comput. Surv. 38(3), New York. DOI: 10.1145/1132960.1132963.

Henelius A, Puolamäki K, Boström H, Asker L, Papapetrou P (2014) A peek into the black box: exploring classifiers by randomization. Data Mining Knowledge

Discovery 28(5-6):1503–1529. DOI: 10.1007/s10618-014-0368-8.

Hothorn T, Lausen B, Benner A, Radespiel-Tröger M (2004) Bagging survival trees. Statistics in Medicine 23(1):77–91. DOI: 10.1002/sim.1593.

Huysmans J, Baesens B, Vanthienen J (2006) Using rule extraction to improve the comprehensibility of predictive models. Published via: FETEW Research Report KBI 0612, K. U. Leuven. DOI: 10.2139/ssrn.961358.

Ishwaran H, Kogalur U, Blackstone E, Lauer M (2008) Random survival forests. Annals of Applied Statistics 2(3):841–860. DOI: 10.1214/08-AOAS169.

Johansson U, König R, Niklasson L (1997) Rule Extraction from Trained Neural Networks using Genetic Programming. In: Nonlinear Analysis: Theory, Methods & Applications, ,H. Levent Akin ADA (ed), Istanbul, pp. 1639–1648. DOI: 10.1016/S0362-546X(96)00267-2.

Kliegr T, Bahník Š, Fürnkranz J (2018) A review of possible effects of cognitive biases on interpretation of rule-based machine learning models. arXiv preprint arXiv:1804.02969. URL: https://www.groundai.com/project/a-review-of-possible-effects-of-cognitive-biases-on-interpretation-of-rule-based-machine-learning-models/1.

Li W, Han J, Pei J (2001) CMAR: accurate and efficient classification based on multiple class-association rules. In: Proceedings 2001 IEEE International Conference on Data Mining, pp. 369–376. ISBN: 07-6951-119-8, DOI: 10.1109/ICDM.2001.989541.

Liu B, Hsu W, Ma Y (1998) Integrating classification and association rule mining. In: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, AAAI Press, KDD'98, pp. 80–86. URL: https://dl.acm.org/citation.cfm?id=3000305.

Lu H, Setiono R, Liu H (1995) NeuroRule: A Connectionist Approach to Data Mining. In: Proceedings of the International Conference on very large Databases, ,P. M. D. Gray ,S. Nishio UD (ed), pp. 478–489. ISBN: 15-5860-379-4.

Lundberg SM, Lee SI (2017) A Unified Approach to Interpreting Model Predictions. In: Advances in Neural Information Processing Systems 30, Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds), Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds). Curran Associates, Inc., pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

Meinshausen N (2006) Quantile regression forests. Journal of Machine Learning Research 7:983–999. URL: http://www.jmlr.org/papers/volume7/meinshausen06a/meinshausen06a.pdf.

National Heart Foundation of Australia (2016) Guidelines for the diagnosis and management of hypertension in adults. Melbourne. DOI: 10.5694/mja16.00526.

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12:2825–2830.

Ribeiro MT, Singh S, Guestrin C (2016) "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, ACM, KDD ’16, pp. 1135–1144. ISBN: 978-1-450342-32-2, DOI: 10.1145/2939672.2939778.

Saxena P NA Prakash A (2011) Efficacy of 2-hour post glucose insulin levels in predicting insulin resistance in polycystic ovarian syndrome with infertility. Journal of Human Reproductive Sciences 4(1):20–22. DOI: 10.1016/j.jfma.2016.02.001.

Smith JW, Everhart JE, Dickson WC, Knowler WC, Johannes RS (1988) Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In: Proceedings of the Symposium on Computer Applications and Medical Care, IEEE Computer Society Press, pp. 261–265.

Strobl C, Boulesteix AL, Kneib T, Augustin T, Zeileis A (2008) Conditional variable importance for random forests. BMC bioinformatics 9(1):307. DOI: 10.1186/1471-2105-9-307.

Tan PN, Steinbach M, Kumar V (2005) Introduction to Data Mining, (First Edition). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. ISBN: 03-2132-136-7.

Thabtah F, Cowling P, Peng Y (2005) MCAR: Multi-class classification based on association rule. In: The 3rd ACS/IEEE International Conference on Computer Systems and Applications, 2005., p. 33. ISBN: 07-8038-735-X, DOI: 10.1109/AICCSA.2005.1387030.

United States Department of Health and Human Services (HHS) (2016) Risk Factors for Type 2 Diabetes. URL: `https://www.niddk.nih.gov/health-information/diabetes/overview/risk-factors-type-2-diabetes`.