Local interpretability of random forests for multi-target regression*

Avraam Bardos[1] [0000−0003−1785−1488], Nikolaos Mylonas[1] [0000−0002−5733−543X], Ioannis Mollas[1] [0000−0002−7765−7903], and Grigorios Tsoumakas[1] [0000−0002−7879−669X]

Aristotle University of Thessaloniki, Thessaloniki 54632, Greece
{ampardosl,myloniko,iamollas,greg}@csd.auth.gr

Abstract. Multi-target regression is useful in a plethora of applications. Although random forest models perform well in these tasks, they are often difficult to interpret. Interpretability is crucial in machine learning, especially when it can directly impact human well-being. Although model-agnostic techniques exist for multi-target regression, specific techniques tailored to random forest models are not available. To address this issue, we propose a technique that provides rule-based interpretations for instances made by a random forest model for multi-target regression, inspired by a recent model-specific technique for random forest interpretability. The proposed technique was evaluated through extensive experiments and shown to offer competitive interpretations compared to state-of-the-art techniques.

Keywords: Interpretable Machine Learning · Local Interpretability · Rule-based Interpretability · Random Forest · Multi-target Regression

1 Introduction

The problem of multi-target regression, which has gained popularity in the last few years, involves predicting two or more continuous values, similar to multi-label classification [16]. Given an input dataset with features and corresponding target outputs, the goal is to estimate the target vector based on the input. Multi-target regression has several applications such as air quality monitoring [8], water quality prediction [15], and electronic health record analysis [13].

There are two main approaches to multi-target regression: problem transformation and algorithm adaptation. Problem transformation involves transforming a multi-target problem into multiple independent single-target problems, while algorithm adaptation adapts a specific method to handle multi-target regression data. Techniques falling into the latter category include predictive clustering trees (PCTs) [7], fitted rule ensembles (FIRE) [1], and random forests (RF) [2].

* The research work was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "First Call for H.F.R.I. Research Projects to support Faculty members and Researchers and the procurement of high-cost research equipment grant" (Project Number: 514)

Interpretability provides reasoning behind the decisions of machine learning models. Most machine learning applications benefit from interpretability, especially when they can affect human life directly (health applications) or indirectly (economic applications) [6]. Interpretability in the context of multi-target regression is a scarcely explored topic, with very few techniques being available for this task. PCTs are inherently interpretable, similar to traditional decision trees. Furthermore, FIRE is also inherently interpretable due to the small number and length of the rules it produces. On the other hand, RF is not inherently interpretable, and model-specific interpretability techniques for multi-target regression have not yet been developed.

Various techniques that are model-agnostic provide rule-based interpretations for single-label classification and regression tasks. Such techniques include RuleFit [4], Anchors [14], and LORE [5]. However, they cannot be used for multi-target regression. In addition, a local model-agnostic technique called MARLENA [12] has been designed specifically for multi-label learning problems.

This work addresses the challenge of interpretability of RF in the context of multi-target regression, by introducing a technique designed to explain predictions of specific instances, called XMTR (eXplainable Multi Target Regression random forests). Our technique is based on a recent technique for local model-specific interpretability of RF, called LionForests (LF), which can accommodate both single-label and multi-label tasks [9,11]. XMTR provides rule-based interpretations for the predictions of the underlying model. To evaluate our method, we conducted quantitative, qualitative and scalability experiments, which demonstrate that XMTR offers competitive interpretations to those of state-of-the-art techniques. Overall, our results show that XMTR is a practical solution for enhancing the interpretability of RF in multi-target regression tasks.

2 Explainable multi-target regression

This work contributes a technique for addressing the interpretability of RF models for multi-target regression by proposing an extension to LF. First, we will present some fundamental concepts necessary for both LF and XMTR. Then, we will present the main three steps of LF, which include: a) path extraction, b) path reduction, and c) rule composition, and the extensions we are introducing.

LF provides a single rule, which explains the decision of a particular instance, without affecting the structure of the model. Through path reduction, it is possible to reduce the number of features in the rule, as well as to widen the feature ranges, making the interpretation more comprehensive to the end user. A rule provided by LF is formulated as a set of feature ranges. An example of such a rule is "if $3 \leq f_2 \leq 8$ and $7 \leq f_3 \leq 12$ then t_i", where a model predicted target t_i, based on features f_2 and f_3, among the available ones.

Path conclusiveness. Conclusiveness, is a vital property of rules [9]. A rule is conclusive if the model's prediction being interpreted for an instance remains the same, both when the values of the excluded features (f_1 and f_4 in our example) are modified arbitrarily and when the values of the included features (f_2 and f_3 in our example) are modified within the specified ranges.

Allowed error. The concept of *allowed error* [9], enables the formation of conclusive rules in tasks like regression. With *allowed error*, it is possible to reduce the number of paths and form a smaller rule, while guaranteeing a predictive error within the allowed limit. The default value for *allowed error* is equal to the mean absolute error of the model, according to an evaluation procedure (e.g. cross-validation), while it can also be user-defined.

Local error. *Allowed error* will be compared to a *local error*, *local error* $=$ *mae*(*preds*, *r_preds*). The predictions made by all the trees are *preds*. The *r_preds* refers to either the prediction made by an individual tree when its path is included, or the highest/lowest possible prediction that the tree could make (highest/lowest value among all the tree's leaves) when its path is excluded, depending on which value is farthest from the actual prediction. If we have an RF with $|T|$ trees and if we exclude $|E|$ trees by choosing to keep only $|K| = |T| - |E|$, the final rule may not account for the decisions made by the excluded trees $|E|$. If a change is made to a feature that is not included in the final rule, it can lead to a non-conclusive rule. Thus, the final prediction should be adjusted as follows: $p'(x) = \frac{1}{|T|}(\sum_{k \in K} p_k(x) + \sum_{e \in E} p_e(x))$, where $p_k(x)$ is the actual prediction of the tree k and $p_e(x)$ is the minimum/maximum prediction of the tree e.

Path extraction. The initial step is to perform a *path extraction* procedure, which aims to collect all the paths taken by the trees in the RF that the given instance follows to obtain its predicted values. During this process, each tree is traversed, and the path that the instance takes from the root to the leaf node to arrive at its prediction is tracked. While following each path, the decisions of each internal node are examined, and the feature used for the split at that node is recorded, along with its threshold value. Hence, for each tree, a path is produced that includes all the features that were used, as well as their minimum and maximum threshold values, and the predicted value of the instance.

Path reduction. Once the features and their ranges for each path are gathered, the subsequent crucial step is *path reduction*. The purpose of this step is to exclude paths that are not essential to the decision. In [9], several reduction strategies are proposed, but here we focus solely on association rules reduction since it is suitable for regression tasks. Association rules are utilized to assign a confidence score to each path, and the features with lower confidence scores are gathered in an empty feature set. The iterative process of enriching the feature set continues until the number of paths, whose conjunctions of features are all included in the feature set, produces a *local error* smaller than the *allowed error*.

Rule composition. The last step is to combine all knowledge acquired from the earlier steps, and create a single rule expressed in natural language. In the *Rule composition* step, for each feature, we collect its ranges from the reduced paths in which it appears, and afterwards, we compute the intersection of its ranges. A conclusive rule contains these intersections of all features.

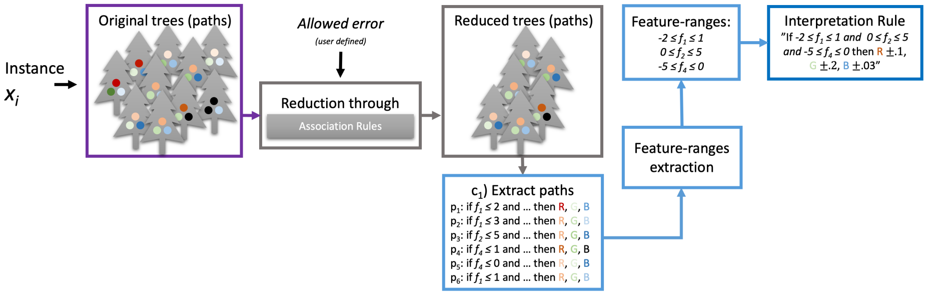
Instance
x_i
Original trees (paths)
Allowed error
(user defined)
Reduction through
Association Rules
Reduced trees (paths)
Feature-ranges:
$-2 \leq f_1 \leq 1$
$0 \leq f_2 \leq 5$
$-5 \leq f_4 \leq 0$
Interpretation Rule
"If $-2 \leq f_1 \leq 1$ and $0 \leq f_2 \leq 5$ and $-5 \leq f_4 \leq 0$ then R ±.1, G ±.2, B ±.03"
Feature-ranges extraction
c_1) Extract paths
p_1: if $f_1 \leq 2$ and … then R, G, B
p_2: if $f_1 \leq 3$ and … then R, G, B
p_3: if $f_2 \leq 5$ and … then R, G, B
p_4: if $f_4 \leq 1$ and … then R, G, B
p_5: if $f_4 \leq 0$ and … then R, G, B
p_6: if $f_1 \leq 1$ and … then R, G, B

Fig. 1. Workflow of the proposed approach

Adaptation to multi-target regression. This work extends LF towards interpreting multi-target regression RF models. Our extension, named XMTR, provides a single rule explaining the whole target set. One such example rule, can be seen at the end of the workflow of Fig. 1. Among the path reduction strategies of LF, we selected association rules, for its ability to reduce effectively both paths and features. This is crucial because the generated rules will only contain the necessary features. Secondly, it is the only strategy that can be modified to work for multi-target regression tasks. The goal is to produce a rule that explains the entire prediction, rather than a single specific target at a time.

XMTR modifies the selected path reduction strategy to consider multiple allowed errors, one for each target. Then the local errors are compared based on two schemes. One compares their average to a globally shared allowed error, while the other uses multiple allowed errors, each tailored to a target. The former enables faster and simpler rule production, as the user defines only one error threshold. The latter provides greater flexibility, as the user can prioritize certain targets by setting a less strict allowed error for less important targets and a stricter one for more important ones, according to his/her preferences.

3 Experiments

In this section, we first present the setting of our experiments, including a summary of the datasets we examined, the metrics we used and the competitors we compared. We then present the results of both the quantitative and qualitative experiments we conducted. Finally, we include a scalability analysis, based on the *allowed error* parameter of XMTR. The datasets as well as the code for our technique and experiments are available in our XMTR repository `https://github.com/intelligence-csd-auth-gr/XMTR`.

3.1 Experimental setup

The datasets used are Slump [15], Andro [15], and Facebook metrics [10], all of them concerning multi-target regression. These datasets have $\{103, 49, 500\}$ examples, $\{7, 30, 18\}$ features, and $\{3, 6, 4\}$ targets. No pre-processing took place besides for Facebook metrics, where the features {like, Share, Paid} were excluded due to containing missing values, and one categorical feature was excluded, as this work does not address this issue.

We employed global (GS) and local (LS) decision tree surrogates, along with MARLENA (MA). As MARLENA was originally intended for multi-label interpretability, we had to make some adjustments to make it suitable for multi-target regression. In specific, we converted the surrogate model of the technique into a regressor, and modified the rules exported to include its prediction. More details regarding these techniques can be found in our repository.

We used three evaluation metrics. The first is *coverage*, which quantifies the amount of instances covered by a rule. Higher values indicate better performance. The second is *rule precision*, which quantifies the number of correctly covered instances. Lower values indicate better performance, as we measure MAE. Finally, *rule length* quantifies the number of conjunctions present inside the rule. Depending on the user and case, lower or higher values may be preferred [3].

3.2 Quantitative results

The RF models' parameters were selected through a grid search process, and their values can be located in our repository at XMTR. However, we report the RF models' performance in MAE for each dataset; 1.7311 for Slump, 1.3326 for Andro, and 0.1008 for Facebook metrics. Our quantitative experiments also involve three distinct *allowed error* values for each dataset, which determine the maximum permissible amount of error for each rule concerning the target values. These *allowed error* values are denoted with #1, #2, and #3 and correspond to $0.2, 0.25$, and 0.5 for Slump, $0.55, 0.6$, and 0.7 for Andro, and, finally, $0.1, 0.3$, and 0.5 for Facebook metrics, respectively. While a plethora of *allowed error* values were tested, this paper showcases only the ones that exhibit noticeable changes in the performance of MAE and rule length metrics, due to space limitations.

The data is partitioned using a 10-fold cross-validation strategy, and the training set is employed to initialize XMTR and its competitors. For every test instance, we generate interpretations and compute the metrics. Finally, we average these metrics across all instances, obtaining the final value.

Table 1 presents the results. In terms of coverage, GS and LS demonstrated relatively higher performance compared to XMTR and MA, suggesting that the rules generated by the latter two techniques are more specific to the instances. Regarding MAE, XMTR showed competitive performance across all *allowed error* categories, achieving comparable results to the other techniques. However, for #3, which corresponds to higher *allowed error* values in all three datasets, XMTR's performance was slightly lower. This outcome is expected, as allowing a higher error value tends to lead to more general rules with larger errors.

LS produced the shortest rules, while XMTR generated longer ones due to its higher specificity. Furthermore, higher *allowed error* values resulted in shorter rules, reflecting the greater path reduction mentioned earlier.

	Coverage			MAE			Rule Length		
Technique	Slump	Andro	Facebook	Slump	Andro	Facebook	Slump	Andro	Facebook
XMTR (#1)	0.09	0.13	0.04	0.09	0.06	0.00	6.8	21.4	7.9
XMTR (#2)	0.09	0.17	0.04	0.17	0.25	0.02	6.7	20.4	7.6
XMTR (#3)	0.10	0.20	0.05	1.46	0.68	0.28	4.9	18.1	5.6
GS	0.20	0.35	0.19	0.22	0.34	0.02	3.9	4.3	3.6
LS	0.23	0.34	0.19	0.30	0.40	0.03	2.9	3.5	2.7
MA	0.11	0.23	0.09	0.16	0.21	0.02	4.8	6.9	4.7

Table 1. Results of the quantitative experiments comparing XMTR, GS, LS, and MA, regarding Slump, Andro, and Facebook metrics datasets, across the coverage, MAE, and rule length metrics

3.3 Qualitative results

We randomly selected an instance from the Slump dataset to conduct qualitative experiments, presenting the XMTR and MA rules. Slump was selected due to the small number of features, allowing for better visualization. For the selected instance, the RF model predicted 7.9 for Slump, 7.2 for Flow, and 4.9 for Compressive_Strength targets.

> ***XMTR:*** *if* $207 \leq Water \leq 208$ *&* $100 \leq Slag \leq 107$ *&* $708 \leq Coarse\_Aggr \leq 753$ *&* $137 \leq Cemment \leq 151$ *&* $126 \leq Fly\_ash \leq 141$ *then* $Slump : 7.9^{\pm 0.8}$*,* $Flow : 7.2^{\pm 0.7}$*,* $Compressive\_Strength : 4.9^{\pm 0.2}$

> ***MA:*** *if* $185 < Water \leq 209$ *&* $Slag \leq 144$ *&* $Coarse\_Aggr \leq 843$ *&* $80 < Fly\_ash \leq 198$ *&* $Fine\_Aggr \leq 885$ *then* $Slump : 7.7$*,* $Flow : 6.8$*,* $Compressive\_Strength : 4.8$

Both rules covered 5 out of 7 features. XMTR excluded SP and $Fine\_Aggr$, while MA excluded SP and $Cemment$. Changing SP did not affect the prediction, but modifying $Cemment$ from 145 to 136 (1 lower than XMTR's suggested range) did have an impact. Conversely, adjusting $Fine\_Aggr$ from 883 to 640 (the minimum value in the training set) did not affect the prediction. Therefore, XMTR correctly excluded the two features that don't affect the prediction, while MA mistakenly excluded $Cemment$, which seems to impact the target. We excluded LS and GS from this analysis to keep the paper concise. However, both LS and GS exhibited similar behavior to MA. This experiment aimed to highlight the inconclusive nature of MA compared to XMTR. If even one rule is inconclusive, the conclusiveness property doesn't hold for that technique. On the other hand, XMTR, which builds upon LF, maintains that property.

3.4 Allowed error and scalability

We aim to measure the effect of *allowed error*, in the time response of XMTR. We anticipate a consistent increase in the time for rule generation in correlation to the value of *allowed error*. Higher error values allows for greater path exclusion by XMTR, resulting in higher time response. We use two synthetic datasets created through *scikit-learn*, one relatively small (D1) containing 500 instances, 10 features and 5 targets, as well as a larger one (D2) with 5000 instances, 50 features and 7 targets. Furthermore, the RF model we used for this experiment consisted of 500 estimators, and we utilized 5 different *allowed error* values.

	D1		D2	
allowed error	Time	#Paths	Time	#Paths
0.05	0.93	482	0.58	495
0.1	1.89	466	1.03	488
0.15	2.73	449	1.30	481
0.2	3.61	436	1.75	473
0.25	4.38	418	2.01	466
0.3	5.22	401	2.57	457

Table 2. Scalability analysis of XMTR

Table 2, presents the time required by XMTR to produce the rules, as well as the number of paths used for that rule out of 500 initial ones. It is visible through these results that our hypothesis holds true and higher error values allow for greater reduction and thus higher time responses. It is also important to mention that the total number of instances (500, 5000) did not have any significant influence on the time performance. Therefore, the choice of *allowed error*, is the most critical to the time response of our method.

4 Conclusions

Our work tackled the challenge of interpretability in multi-target regression tasks for RF, which is a scarcely explored topic with important applications. We introduced a technique called XMTR, which modifies the existing LF approach for local model-specific interpretability of RF. Through quantitative and qualitative experiments, we demonstrated that XMTR outperforms current state-of-the-art techniques. Additionally, through a scalability analysis, we presented how XMTR's *allowed error* parameter affects time response. Our research highlights the potential of XMTR in multi-target regression tasks and contributes to the advancement of interpretability research in this scarcely explored area.

References

1. Aho, T., Ženko, B., Džeroski, S.: Rule ensembles for multi-target regression. In: 2009 Ninth IEEE International Conference on Data Mining. pp. 21–30 (2009)

2. Breiman, L.: Random forests. Machine learning **45**(1), 5–32 (2001)
3. Freitas, A.A.: Comprehensible classification models: A position paper. SIGKDD Explor. Newsl. **15**(1), 1–10 (Mar 2014). https://doi.org/10.1145/2594473.2594475, `https://doi.org/10.1145/2594473.2594475`
4. Friedman, J.H., Popescu, B.E.: Predictive learning via rule ensembles. The annals of applied statistics pp. 916–954 (2008)
5. Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., Giannotti, F.: Local rule-based explanations of black box decision systems. arXiv preprint arXiv:1805.10820 (2018)
6. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D.: A survey of methods for explaining black box models. ACM computing surveys (CSUR) **51**(5), 1–42 (2018)
7. Kocev, D., Vens, C., Struyf, J., Džeroski, S.: Tree ensembles for predicting structured outputs. Pattern Recognition **46**(3), 817–833 (2013)
8. Masmoudi, S., Elghazel, H., Taieb, D., Yazar, O., Kallel, A.: A machine-learning framework for predicting multiple air pollutants' concentrations via multi-target regression and feature selection. Science of The Total Environment **715**, 136991 (2020). https://doi.org/10.1016/j.scitotenv.2020.136991
9. Mollas, I., Bassiliades, N., Tsoumakas, G.: Conclusive local interpretation rules for random forests. Data Mining and Knowledge Discovery **36**(4), 1521–1574 (2022)
10. Moro, S., Rita, P., Vala, B.: Predicting social media performance metrics and evaluation of the impact on brand building: A data mining approach. Journal of Business Research **69**(9), 3341–3351 (2016). https://doi.org/10.1016/j.jbusres.2016.02.010
11. Mylonas, N., Mollas, I., Bassiliades, N., Tsoumakas, G.: Local multi-label explanations for random forest. In: Machine Learning and Principles and Practice of Knowledge Discovery in Databases. pp. 369–384. Springer, Cham (2023)
12. Panigutti, C., Guidotti, R., Monreale, A., Pedreschi, D.: Explaining multi-label black-box classifiers for health applications. Precision Health and Medicine: A Digital Revolution in Healthcare pp. 97–110 (2020)
13. Poulain, R., Gupta, M., Foraker, R., Beheshti, R.: Transformer-based multi-target regression on electronic health records for primordial prevention of cardiovascular disease. In: 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). pp. 726–731 (2021). https://doi.org/10.1109/BIBM52615.2021.9669441
14. Ribeiro, M.T., Singh, S., Guestrin, C.: Anchors: High-precision model-agnostic explanations. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
15. Spyromitros-Xioufis, E., Tsoumakas, G., Groves, W., Vlahavas, I.: Multi-target regression via input space expansion: treating targets as inputs. Machine Learning **104**, 55–98 (2016)
16. Waegeman, W., Dembczyński, K., Hüllermeier, E.: Multi-target prediction: a unifying view on problems and methods. Data Mining and Knowledge Discovery **33**, 293–324 (2019)