Master Thesis

# Does interpretability necessarily sacrifice accuracy?

How to improve decision trees and benefit from their interpretability

**MSc Data Science and Marketing Analysis**


**Name** Filip Baumgartner
**Student Number** 620918

**Coach** Eran Raviv, PhD
**Co-reader** prof.dr. D Fok

**Date** 19 August 2022

# Table of Contents

# 1 Introduction

The aim of the master thesis is to empirically test the trade-off between accuracy and interpretability of white-box machine learning models, namely decision trees, compared to more complex models, also called black-box models, such as Random Forest. The main scope of the work is regression, not classification. Naturally, it is expected that decision trees will perform worse than Random Forest models in the majority if not all cases on used data sets, however, this comparison may provide a clear picture of the crucial problem – the sacrifice of accuracy in order to obtain a simpler and transparent model, that has clear interpretation, and its outputs are very well explainable to the audience. In other words, we will discover how much performance from Random Forests is left on the table in order to rather deploy simpler models, that enable a better understanding of the processes in obtaining their outputs and so may meet the expectations of stakeholders, decision-makers, and regulator more satisfying than black-box models.

The current trend in development and deploying of machine learning and artificial intelligence solutions is tremendous and impact not only all areas of businesses but also everyday lives. This causes decision-makers to rely on making decisions on data-related solutions and outputs and additionally, these systems have the power to influence people's behavior and their decisions, for example in the purchasing processes or trust. However, humans often don't understand why AI systems make specific decisions or behave in certain ways. This obscurity can undermine users' trust in the system, particularly in situations where the stakes are high, and lead to system rejection (Ray, 2020). That implies, there are many fields that calls for interpretable machine learning system. Crucial areas where interpretable AI may play a crucial role is healthcare, justice, finance and other regulated domains that leverage ML for important predictions with impact on human's lives (Rudin, 2018).

As stated by (Russell n.d, p. 11*): "Machines are beneficial to the extent that their actions can be expected to achieve our objectives."* Hereby, the question states that what should underline the trust that actions based on models achieve given objectives? The crucial factor may be transparency and clear interpretability of interpretable models. Every time a predictive algorithm is evaluated it can be done in two ways: Is it enough for stakeholders to know just what is predicted or is it important to know why the prediction was made in each way?

There are many machines learning models and solutions, some of them are very transparent and interpretable and some of them less. The main difference between difficult-to-explain black-box models and interpretable (white-box) models is that white-box models provide clear explanations of how they behave, how they produce predictions and what factors influence these predictions to what extent. Typical examples of white-box models are linear models that are very intuitive for human's understanding, such as linear regression or decision trees, that remain transparent all the time and are also pretty much intuitive if only a reasonable number of predictors are included in the model. In the contrast, black-box models are

highly non-linear, and users can only observe the input-output relationship, when predictions are made based on provided inputs, however, there is no transparency in the behavior of these models, and it is tricky to discover any exact conditions and reasons why a prediction was made and what led a model to obtain given results.

This problematic side of black-box models may lead to, already mentioned, undermining trust in these models. Users, such as decision-makers, may be more tempted to trust models that are transparent with their outputs, so they can have a clear perception of why those models make predictions, which may positively influence their confidence in using related models.

Naturally, there can be a trade-off between interpretability and accuracy of models, when more advanced techniques (black-boxes) may perform better than white-box models, which may lead to sacrifice of performance in order to remain interpretability, which may not be in some cases bad approach. However, as proved for example by the research of Rudin (2019), this does not always need to be a case and white-box models may perform as good as black-box models, however, it requires additional time determination and deployment of other techniques in order to obtain a well-performing white-box model. Thus, it is obvious there is a demand for transparent white-box models, that can perform relative well.

### *1.1 Importance of Interpretations and its Relevance for Machine Learning*

Interpretation can be often really fruitful and even necessary, because as stated by Doshi-Velez and Kim (2017) *"the problem is that a single metric, such as classification accuracy, is an incomplete description of most real-world tasks"* however there are cases when it is not needed at all and model producers do not need to take care of it. Doshi-Velez and Kim (2017) argue that explanations are not necessary when there are no serious consequences for unacceptable results or when we can trust a system's decision because it has already been well studied and tested in real-life applications. Anyway, there is always a reason to be aware and suspicious of errors generated by machine learning systems, as there are proven cases in different domains where machine learning failed and caused serious consequences. For example, the cases of incorrectly denied people (Wexler, R., 2017), wrong bail decisions that led to the release of dangerous criminals or more cases in domains of healthcare and finance (Varshney, K.R., 2017). In the case of finance, there is even serious evidence that wrong assumptions in modelling played role in the mortgage crisis in the USA (Donnelly, C., 2010). Doshi-Velez and Kim (2017) outlined the factors that might be optimized by means of interpretability:

- Fairness – makes sure that outputs are objective and do not implicitly or openly target protected groups for discrimination.
- Privacy - Guarantee that sensitive data is adequately safeguarded.
- Reliability/Robustness - Ensure that modest changes to the input do not significantly affect the forecast.

- Causality – Only causal connections are identified.
- Trust - Humans are more likely to trust a system that explains its judgments as opposed to a black box that simply produces the decision.

Naturally, ML algorithms can have different aims, such as saving as many lives as possible in healthcare or minimizing credit defaults in banking. Despite this, the above-mentioned requirements represent key downstream tasks that one may wish to optimize while still pursuing the system's ultimate objective. These downstream tasks could be part of the issue definition; therefore, they must be considered while developing the system, which is impossible without interpretability (Diogo V. Carvalho et al., 2019).

However, as argued by Wang et al. (2018) - interpretability does not mean automatically credibility. Regarding the author, a credible model must be not only interpretable, but its predictions should be in line with well-established domain knowledge and is not worse than other models in terms of performance.

### *1.2 Research questions*

*1. What are the differences in performance between Decision Tree and Random Forest on various datasets and to what extent is sacrificed accuracy acceptable in order to remain interpretability?*

*3. May the system of a few merged decision trees with given weights of their outputs outperform a single decision tree and achieve accuracy closer to the accuracy of Random Forest, while the explainability of this model remains?*

*4. What are the best parameters and properties of our new model, in terms of the number of merged decision trees and the distribution of weights that are given to outputs of different trees? Does the type and structure of datasets play a significant role?*

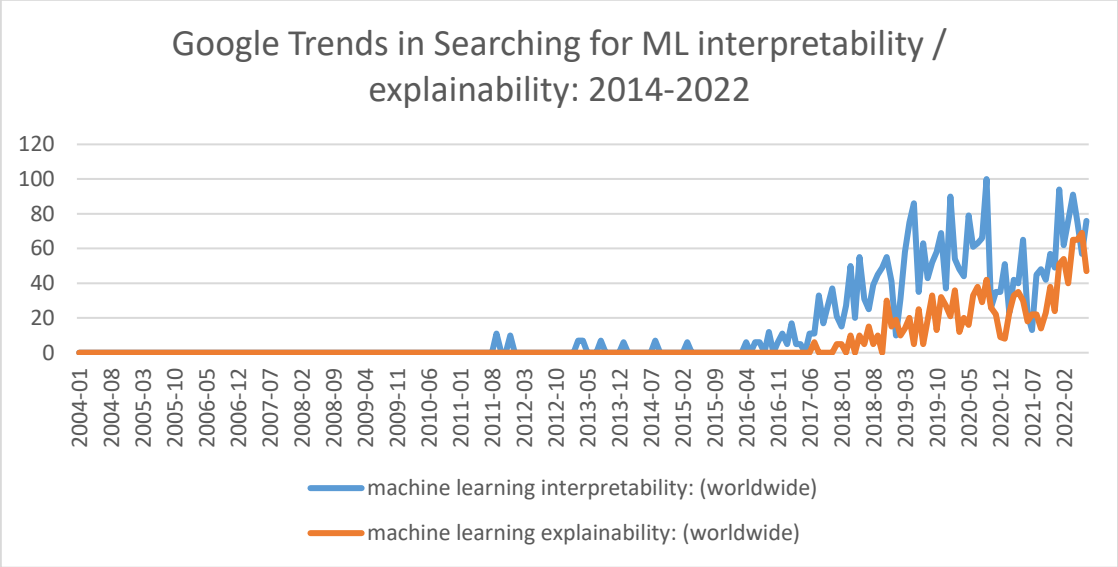
Importantly, to contribute to this hot topic of interpretability and accuracy trade-off, a new method and approach to constructing the white-box model will be introduced. This method should satisfy a condition of interpretability and remain very transparent, however, it should reach – at least in some cases – better performance than a single decision tree.

### *1.3 Academic Relevance*

The topic is relevant for various parties that are influenced by machine learning models. To deal with the transparent model is not relevant only for decision-makers and for regulatory purposes, but it is also trendy due to new legislations that constrain data privacy of individuals, such as GDPR, which gives people the right to an explanation of the algorithm’s decision that affects one’s life (Wachter, S.; Mittelstadt, B.; Floridi, L., 2017). As stated by Doshi-Velez and Kim (2017) , there are ways how to explain black-box model, such as post-hoc explanations, however these methods are just simplifications and additional

estimates of model's behavior and do not need to be reliable and tend to be misleading (Rudin, 2019). Thus, for a reliable understanding from where predictions come from it is often more fruitful to construct fully interpretable machine learning.

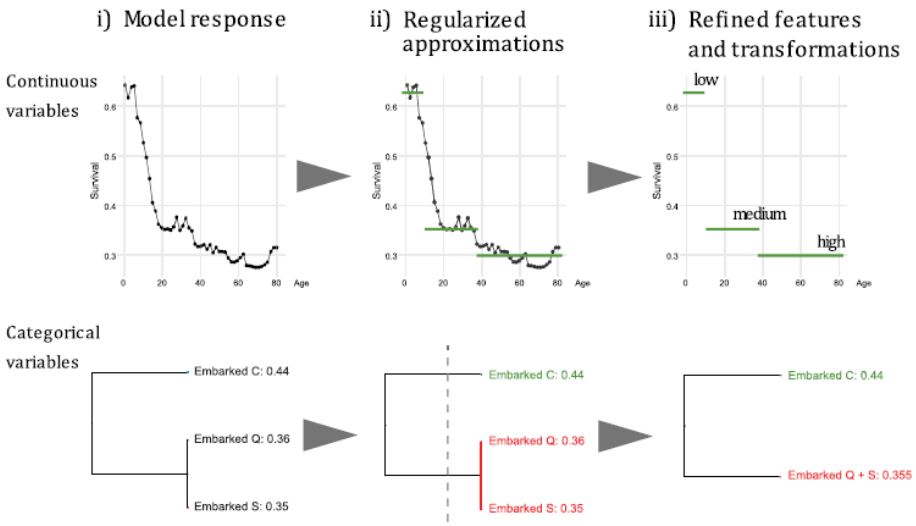
Hereby, we can see a graph that shows a growing trend of interest in searching for machine learning interpretability or explainability keywords at Google. It is clear that interest in these topics is increasing since 2016 and the topic is very relevant nowadays in all aspects and areas.



## 2 Related Work

Gosiewska, A. Kozak, P. Biecek (2021) introduced a framework based on Supervised Assisted Feature Extraction for Machine Learning framework (SAFE ML). The principle of SAFE ML is to construct an accurate supervisor model of high complexity that does not have to be interpretable in order to extract information about variables transformation that can be used in further feature engineering in process of building an interpretable model. Thus, the main point is to obtain additional information via the black-box model about variables and then to transfer this information into the white-box model via the creation of new features to provide a hint about discovered information hidden in data that could not be obtained via white-box due to its limitations. In this case, information from black-box for continuous variables are extracted via calculation of Partial Dependence Profile for given feature. Then, Pruned Exact Linear Time method is used to categorize continuous variable via binning Partial Dependence Profile of given variable (R. Killick, P. Fearnhead, I.A. Eckley, 2012). The variable is discretized in the changepoints of the highest variability and the total number of changepoints depends on the penalty term. These new features together with original features are then used as inputs in selected white-box model.

For categorical features, hierarchical clustering is deployed, when similar prediction of black-box algorithm is merged, and then hierarchical clustering is used so new information about similar clusters can be provided into white-box model as new feature.




*Source: Gosiewska, A. Kozak, P. Biecek (2021)*

Zeng, Ustun, Rudin (2015) focused on model with high impact on lives – recidivism prediction - and so the main aim was not only to construct accurate model but mainly transparent and interpretable. They proved that traditional method such as Ridge Regression could perform on imbalanced classification as well as more modern methods such as Stochastic Gradient Boosting or Support Vector Machines. Importantly, they also deployed recent method called Supersparse Linear Integer Models (SLIM) (Ustun,

Stefano Tracà, Cynthia Rudin, 2014) that produced accurate, transparent and interpretable scoring system. The main point of SLIM is to produce classifier that balances accuracy and interpretability, and so optimization problem was formulated in the following way:

$$\max_{f} \; Accuracy(f) + C \cdot InterpretabilityScore(f)$$

$$s.t\; InterpretabilityConstraints\; (f) > 0$$

Using these two processes, the optimization may generate an interpretable classification model: firstly, an interpretability score, which encourages interpretable classifiers via regularization; and secondly, a set of interpretability constraints, which confine classifiers to a user-defined interpretable set.

Chen et al (2018) focused on credit risk models, where black-box models are often deployed but on the other hand, also interpretability may be required. However, rather than they would build a black-box model and explain it afterwards via available explainable-AI methods, they rather aimed to construct a globally interpretable model that is in accuracy close to other black-box models and neural networks. They introduced a „two-layer additive risk model" that is decomposable into subdimensions, with each node in the second layer representing a meaningful subscale and all nonlinearities being observable. Importantly, they provided three kinds of explanations that are simpler than the global model but still fully consistent with the constructed model and so they can be reliable. One of these methods is variable importance which is based on identifying the two most important subscales of the model. After the identification of subscales, the most important factors that contribute to the final predictions are identified within each subscale. The second method of explanation a is a consistent rule-based explanation that summarizes a wide range of patterns of the model with respect to the data. Finally, an online visualization tool that allows users to explore the global model and its explanations was constructed.

Wang et al. (2018) focused on model credibility which implies interpretability. They proposed a regularization penalty called „expert yielded estimates (EYE)", which combines expert information regarding well-established correlations between variables and the result of interest, as authors argue that a credible model should be not only interpretable but its prediction should be also in line with well-studied domain knowledge. This domain knowledge is utilized to drive the model's selection of highly correlated variables while advocating sparsity. It was shown that linear regression models that deployed EYE achieved higher credibility than models that used other regularization penalties, such as LASSO and so.

Wei at al. (2021) tried to implement a white-box algorithm that would sufficiently forecast natural gas daily consumption on high dimensional and large samples. Firstly, they introduced weighted parallel model architecture (WPMA) strategy that decreases the number of subseries and their non-stationarity by incorporating k-means clustering and weighting the subseries forecasts for out-of-sample forecasting.

When WPMA is combined with reduction dimension performed by Principal Component Analysis (PCA) and then deployed into Multiple Linear Regression – sufficient white-box model is obtained with a reduction of mean absolute percentage error (MAPE) by 40 % compared to single Multiple Linear Regression. This high performance was comparable to deep learning methods that were tested as benchmarks.

Naturally, there are many solutions based on or related to decision trees as the concept of decision trees exist for decades. Tong et al. (2003) introduced algorithm that merge best performing decision trees for classification purposes with satisfying results of improved accuracy compared to single DT. Their system was built on constructing DTs with predictors that has not been used yet in the previous trees and take into merging only final DTs, that performed better than the very first one DT built on all predictors available in an original dataset.

### *2.1 Interpretability or explainability?*

There are two important terms in the field of current machine learning and artificial intelligence development, that are often used together – interpretability and explainability (Adadi, A., Berrada, M., 2018), however, there are proofs that those can be taken as distinct concepts that can be defined by different definitions and meanings (Broniatowski, D., A., 2021). Interpretation or interpretability can be characterized in several ways. Interpretation can be referred as a human's capacity to make sense of a stimulus (such as the output of a ML model) in order to make a decision (Kintsch, W., 1974). Following the statement, Broniatowski, D., A. (2021) defines an interpretable machine learning model as a model that can provide users with a description of what a stimulus (in this case a datapoint or model output) means in context. Similarly, Miller (2017) defines interpretability as *„the degree to which a human can understand the cause of a decision "*. Doing so, interpretability can help humans to get better insights and so to improve high-level decision-making.

On the other hand, explanations rather look for descriptions of processes or rules that were implemented in order to obtain an outcome independent of context (Broniatowski, D., A., 2021). The typical example of explanation is outlying how a model obtained the results, or in a broader context, as stated by Miller (2017) – explanations are answers to why–questions. The justification for the explanation of an algorithm's output is based on the implementation or technical procedure utilized to obtain that output. In contrast, an interpretation is justified according to the algorithm's functional goal (Broniatowski, D., A., 2021).

To put these two definitions into an example and paint their difference, we can take the traditional example of a prediction task – to predict whether a student will succeed when applying for university or not. The main point is to build an algorithm that would classify students into two groups – successful and

unsuccessful based on some characteristics. The purpose of interpretability would be to clearly see what determines the success of candidates and then based on these rules it is possible to evaluate the quality of the model and see if it is not biased or racial, which would mean trouble. On the other hand, the goal of the explanation would be to describe how outputs of models were achieved and how a model decided to make given decisions, so we would talk more about the technical side of a model than about the rationale side and its context behind given predictions.

Additionally, C. Rudin (2019) defines understandable models as those models that requires additional models or features in order to provide explanations to stakeholders, while interpretable models are able to provide explanations without a need to use any additional techniques.

### *2.2 Black-box vs White-box models*

Regarding interpretability and explainability, machine learning algorithms may be classified into two common groups, that describe their level of human understanding and possibilities to interpret them. The first group are so-called white-box models. A White-Box model is one whose underlying logic, operations, and programming processes are transparent, and whose decision-making process is consequently interpretable (E. Pintelas, I. E., Livieris, P. Pintelas, 2020). The typical examples of white-box models are decision trees or linear models. However, even decision trees or linear models may become uninterpretable. In the case of decision trees, the deeper and more complex decision trees with many leaves the more difficult the interpretability and intuitive explanations begin (Molnar, 2019). For linear models, it may be tricky to interpret models with many interaction terms or relationships that are non-linear. The greater the number of nonlinearities and interactions, the less accurate the linear model and the less credible the explanations (Molnar, 2019). Anyway, these models are more suitable for implications that require a lot of transparency for whatever reason, however they may be not always the best performers. To develop interpretable white-box model with high performance may be challenging task and developers can deal with trade-off between accuracy and interpretability.

Kuhn and Johnson (2013) stated, that *"Unfortunately, the predictive models that are most powerful are usually the least interpretable"*. However, on the other hand, for example Rudin (2019) argues that this accuracy – interpretability trade-off does not always imply, especially if data are structured.

The second group are black-box models. These models are usually models with non-linear and non-monotonic function, which are hard to explain. The most typical examples of these group of models are neural networks or ensembled methods, such as random forest or boosting. These algorithms tend to perform better and more accurate than white-box models, however they are much more complicated for understanding, interpretability, intuition and for proving the background of their predictions that were obtained.

Additionally, there is one sub-group of models that stick somewhere in between. The point of this methodology is to combine both group of models, black and white box, in order to obtain interpretable model (white box) with features generated via black-box model, that support the predictive performance of the white-box.

### *2.3 Types of Interpretability*

There are two approaches how to provide interpretability for machine learning models:

***Intrinsic interpretation methods***. They are interpretable by nature, which means that models are built in the way that they can be interpreted clearly from the beginning. This type of interpretability refers to white and grey box models, which are considered as interpretable in general and there is no need to develop any other solutions in order to explain predictions differently than they were explained by outputs provided by models themselves.

***Post-hoc interpretation methods***. Post hoc interpretation methods provide way how to interpret and explain black-box models and their prediction that were made. In contrast to intrinsic interpretations, these methods were developed externally and are not natural parts of black-box algorithms. The advantage is that these methods can be applied across many models as they do not directly depend on the methods themselves but rather provide universal framework. However, on the other hand, these explanations are still just different algorithms that try to explain other algorithm, which means they are based on other assumptions and their results do not have to be reliable and unbiased.

Additionally, post-hoc interpretation methods can be divided into two separate categories depending on what they aim to explain:

1. Global Model-Agnostic Methods - describe the behaviour of a machine learning model on an average basis. Because these methods describe the general behaviour of the model, they are useful if we want to understand the general mechanism in the data or to debug a model.

2. Local Model-Agnostic Methods – are opposite to Global Methods as they aim to explain individual predictions made by a model. They are useful if we want to see the background of single pred data points and understand how a prediction was obtained and what determined a prediction of given data point.

There are several methods that were used for these purposes and are highly acceptable in the field, some of them are following:

- Permutation Feature Importance – Global Method

- Partial Dependence Plots (PDP) – Global Method
- Accumulated Local Effects (ALE) Plots – Global Method
- Local Surrogate (LIME) – Local Method

***Permutation Feature Importance***. The method was introduced by Breiman (2001) in his Random Forest paper and based on this idea - model-agnostic method generalized for purposes of deployment in other models was introduced by Fisher, Rudin, and Dominici (2018). Permutation Feature Importance is a global method that is used for discovering the importance of features included in a model. Values for each feature in dataset are permuted and the change of model performance is recorded. The more important feature the higher increase in model error, and so it can be assumed that permuted features that led to high increase in error are important for model and its ability to produce accurate predictions. On the other hand, if there was just small change in error so it shows that given features is probably not that important for learning of model and does not provide any sufficient information and thus may be excluded in order to simplify model.

***Partial Dependence Plots***. Partial Dependence Plots (J. H. Friedman 2001) demonstrate the impact of one or even two variables on the predicted result of a machine learning model. It allows to discover the type of relationship between dependent variable and an explanatory variable and so it is possible to see if there is linear, monotonic or more complex relationship. The intuition behind PDP is that all observations of feature of interest (dependent one) must be changed, while other independent features remain unchanged. Observations are changed in the way that they are replaced by observed values in min-max interval step by step. That means that the starting point is to replace all observations of dependent variable with minimum, then to use trained model to score each observation and take an average. In this way, observations are changed by observed values in interval until it reaches maximum. In this way, it allows to plot average prediction for all levels of observed values in dependent variable, which lead to clear and intuitive explanations. However, it brings a few challenges. One of the main and crucial assumptions to obtain reliable PDP is assumption of independence. If the features are correlated, it leads to creation of new data points in the areas of feature distribution, that are very unrealistic. Typical example to demonstrate this issue is height-weight relationship - if height is dependent variable for PDP explanations so in some step we obtain target scores for very unlikely relationship, such as very high height but low weight.

***Accumulated Local Effects (ALE) Plots***. Solution for the problematic assumption of PDP are Accumulated Local Effects Plots (Apley, Daniel W., and Jingyu Zhu, 2020), which are faster and more unbiased – especially in the case of correlated features, when PDP cannot be trusted. While PDP works with marginal distributions and can lead to unreal combinations, so ALE Plots work with conditional distribution and instead of taking averages of prediction they calculate differences. This leads to working with more realistic grid of values (conditional distribution) and avoiding mixing the effect of a feature with the effects of all correlated features (differencing). Thus, ALE Plots should be usually more preferred method to PDPs,

mainly in environment of highly correlated features in dataset. On the other hand, ALE Plots implementation is more complex than PDPs and interpretation is less intuitive compared to average predictions shown by PDPs.

***LIME***. Local interpretable model-agnostic explanations (Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin, 2016) is a method that proposed the implementation of local surrogate models. Surrogate models are interpretable models that are trained to approximate the predictions of black-box models. LIME focuses on explanation of individual prediction of black-box models in the way that surrogate model is trained locally and so we can understand why a model mode a certain prediction. LIME works in the way that dataset is perturbed and black-box model predictions are obtained for datapoints that we want to explain. These new samples are then weighted according to their proximity to the datapoints of interest. Following, weighted interpretable of free choice – usually regression or decision trees – is trained on the dataset with variations and this interpretable model is able to explain certain predictions.

# 3 Methodology

## *3.1 Models*

The idea behind the research is to propose a method that yields the final model with more accurate regression predictions while keeping its interpretability. The base of the method, like several other techniques such as Random Forest, consists of several different regression decision trees. The purpose is to beat the performance of single decision trees while still benefiting from their interpretability and getting as close as possible to the results of Random Forest, which should be in general more accurate than the results of Decision Trees.

The aim is to build as many regression decision trees as several predictors in a given dataset, with the difference that each decision tree has a different starting point (root node), which automatically influences other splits in a decision tree and leads to distinct parent and so also child nodes, leading to as many unique and different decision trees as predictors in a dataset. So, except for the natural split based on the automatic calculation of the node split that would lead to the lowest Sum of Squared Error (SSE), the case of this methodology is to artificially constrain the selection of split so, that for each dataset with N predictors, each of N predictors is used for building a decision tree with N-th predictor as a variable for a node split.

In the following step, each out of the N decision trees is tested on a sample of a dataset that was taken out – the so-called validation dataset – in the way that predictions on the new unseen data are made and Root Mean Squared Error (RMSE) is calculated in the following way, where yi is prediction, y the actual value and n number of observations:

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(y_i - y\ )^2}{n}}$$

RMSE is used as the main metrics to evaluate all models in the research, because the main property of RMSE is that the errors are squared and so much larger weights are assigned to large errors, which should be from our perspective penalized more significantly than smaller errors. Naturally, the use of metric usually depends on the topic and problem but in general to penalize large errors seem to be more rational approach then to use metrics such as MAE with linear scale of error weights. When an error term in the form of RMSE is obtained for each of the N decision trees, they are ordered in the way, so the lower RMSE the higher rank of a decision tree.

Following, up to three best-performing decision trees based on RMSE metrics are chosen. Firstly, the results of the two best decision trees are combined. The merging of results of these two best-performing

decision trees is performed with given weights. The decision about given weights is made so the combination of predictions with chosen weights leads to the lowest RMSE when merged predictions with given weights are compared to the actual values.

In the next step, similarly to the previous procedure – predictions from the three best-performing decision trees are merged with given weights for each prediction of the three trees. It is expected that adding one more tree with a given weight may lead to an improvement of accuracy and so a decrease in RMSE, however, there is already assumption that it does not need to be always the case and improvement may not be present or may be just marginal.

In the case when two trees would be sufficient and merging of the third one does not bring any improvement, so the process is to proceed only with two trees as it would not make sense to work with three trees with the same performance, as two trees are still better interpretable than three and focus on interpretability is one of the main purposes of the research.

As the final step – results of several models are compared to each other and so the decision about deployment of the merged model may be taken. The main purpose is to improve accuracy of single decision tree and get as close as possible to results of Random Forest. To evaluate this, results of following models are compared to the results of the merged decision tree:

pruned decision tree with cross-validated cost-complexity parameter (described above)

Random Forest with parameters tuning based on cross-validated grid search (described above)

If the results of merged decision trees are better than single decision tree so it should be worth a shot, as the interpretability of two and even three merged decision trees with given weights is still pretty much high. In the case of poorer improvement, it can be considered to keep with a single decision tree. However, it is expected that there would be a significant difference between merged trees and random forest performance most of the time, but it is also assumed that in some cases merged trees may get really close to performance of random forest, which would usually depend on type of data.

### *3.2 Methodology Guideline*

The whole process can be briefly summarized in following steps:

1. Train N decision trees with N different starting points (nodes), where N = number of predictors in a dataset. Tuning parameters are given in the way to remain clear interpretability and to do not grow too large and complex trees. We defined not complex tree in the way that it does not have more than 10 branches, which is always checked and it is achieved by higher Complexity Parameter, most of the time at level 0.05.

2. Test performance of each of N decision trees on a validation unseen dataset and calculate RMSE
3. Choose up to three best-performing decision trees with lowest RMSE
4. Look for weights for outputs of all chosen decision trees in the way that combination of given weights for individual outputs would lead to the lowest RMSE of the final prediction
5. Merge results of chosen decision trees with given weights to obtain the final prediction
6. Obtain the final model and results, compared results of merged decision trees with single decision tree and random forest; evaluate improvement and sacrifice of accuracy in order to remain interpretability

### *3.3 Decision Trees*

Decision tree is one of the most traditional supervised learning algorithms invented by JR Quinlan (1986). Decision Trees can be used both for regression when dependent variable is continuous and classification tasks when dependent variable is categorical, which makes from them universal algorithm for many uses. The main principle of decision tree is to predict dependent variable by learning simple decision rules, which basically represent if-else conditions leading to the final predictions that satisfies all given conditions created in the process of growing a tree.

The principle of building decision trees is to stratifies feature space in the way that predictor space, which states for set of possible values for X1, X2...Xn predictors into Z distinct and non-overlapping regions R1, R2...Rn. Then, for each observations that falls into one of the region Ri – prediction is made, which is in the case of regression trees the mean response value for all training observations that were included in the given region Ri during building a tree on training dataset (James, Witten..).

Created regions in a stratified space can have any shape, however the aim is to find regions that minimizes error, which is in the case of regression trees Residual Squared Error (RSS), so the purpose is to minimize variance in each created region R.

$$\sum_{j=1}^{J} \sum_{i \in R_j} \left( y_i - \widehat{y_{R_j}} \right)^2$$

$$where: \widehat{y_{R_j}} = mean\ response\ for\ observations\ in\ j-th\ region$$

Sadly, it is computationally impossible to consider every possible J box partition of the feature space. Thus, top-down greedy approach is taken, also called as recursive binary splitting. It means that splitting of the predictors space starts at the top of the tree, at which all observations belongs to the one region and every split in the process is consider and made as the best split at that particular step, without considering other splits ahead that might lead into better results in following steps.

Recursive binary splitting in performed in the way that firstly, predictor Xi is selected and then the best predictors cut point that would lead into the largest reduction of RSS is chosen. In this way, predictors space is split into two regions with given conditions created by cut point. All predictors are considered in this way and the final predictor with its cut point that leads into the smallest RSS is chosen for the final decision about a split.

For any j and s, we define the pair of half-planes

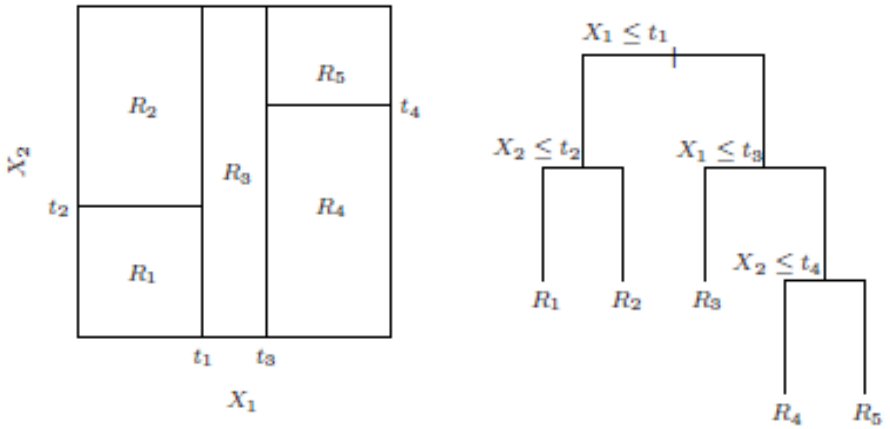
$$R_1(j,s) = \{X \mid X_j < s\} and R_2(j,s) = \{X \mid X_j \geq s\}$$

and we seek the value of j and s that minimize the equation:

$$\sum_{i:x_i \in R_1(j,s)} (y_i - \widehat{y_{R_1}})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \widehat{y_{R_2}})^2$$

$$where: y_{R_1} = mean\ response\ for\ observations\ in\ region\ R1\ and\ y_{R_2} = mean\ response\ for\ observations\ in\ region\ R2$$

The whole procedure as described above is repeated until the final decision tree is built, however in later stages only restricted areas of predictor space are considered for further splitting, as the whole area was already split once. So, for instance, the second step splits are considered separately for region R1 and separately only for space of region R2 and so on.




**Pruning**

If decision tree is built in the way described above in may lead to very complex decision tree with many regions, which may cause good performance on training data but poorer performance on test data – overfitting. Additionally, the more complex tree the more difficult interpretation, as the final predictions may be based on many conditions which can be not only overwhelming but also extensive for purposes of visualizations of outcomes.

To avoid this situation, pruning of decision trees may be deployed and so smaller sub-tree can be obtained. The goal of pruning is to reduce number of regions and so to obtain more general rules and broader regions, while keeping as much accuracy as possible.

There are several possibilities of pruning, one of them is so called pre-pruning, which means that some criteria are set, and a tree is being built until criteria is not met, then it stops. Typical examples of widely used criteria are:

- Minimum split – minimum number of observations that must exist in a node in order for a split to be attempted
- Minimum bucket - minimum number of observations in any terminal leaf node

However, as stated by James, Witten (2013), it is recommended to build complex and deep tree and then to deploy post-pruning to obtain smaller sub-tree, as pre-pruning may lead to situations when important and meaningful split is avoided due to poor split that would forego the important one, but a tree growing would have to be stopped due to criteria that were met by prior poor split.

For these purposes – post-pruning is used and popular method of simplifying of deep decision trees. One of the most widely used and popular method is Cost complexity pruning (Breiman, 1984) The aim is to select less complex sub-tree that would lead to the lowest error. This error can be estimated via cross-validation – which mean splitting training dataset into K-fold, each i-th fold is taken apart of training observations and then used as validation set, so error can be estimated and averaged across all folds. Via this process, various inputs for tuning parameter α can be tested and the one that yields the lowest error chosen for the final model. In the post-pruned sub-tree, the goal is to minimize function

$$\sum_{m=1}^{|T|} \sum_{i:} \left(y_i \in R_m - \widehat{y_{R_m}}\right)^2 + \alpha|T|$$

Where $|T|$ is number of terminal nodes in tree T, $R_m$ is subset of predictor space related to the m-th terminal node, and $\widehat{y_{R_m}}$ is the mean value of observations in space $R_m$ .

The tuning parameter regulates a trade-off between the complexity of the subtrees and their ability to match the training data. The higher tuning parameter α the less complex decision tree and thus likely lower fit on the training data.

Besides Cost complexity pruning, there are other methods that can be used, such as Reduced Error Pruning (Quinlan, 1987), when each node is replaced with its most well-liked class beginning at the leaves. The adjustment is preserved if the prediction accuracy is unaffected. However, as empirically discovered by Mingers (1989), even though in general pruning may improve accuracy of decision trees up to 25 % in

domains with noise and residuals variation, so Reduced Error Pruning is one of the least effective pruning methods as it is very sensitive to the number of classes in data.

As a final note, one of the main advantages of Decision Trees is that they are easily understandable, interpretable and also can be nicely visualized. One of the main disadvantages is that their do not have predictive performance as other more powerful methods and lack robustness.

### *3.4 Random Forest*

Random Forest (Breiman, 2001) represents one of the main algorithms that are based on the principles of decision trees but with important additional improvements than usually lead to significantly improved performance over decision trees.

Random Forest combines two methods that tend to improve decision trees. One of them is Bagging (Breiman, 1994), which is implementation of statistical bootstrapping into decision trees. Bootstrapping is random sampling of observations with replacement, which means that part of observations included in dataset of each tree are repeated, as well as that the size of bootstrapped dataset is not reduced. The main principle is that bagged trees are based on many decision trees, which are built on random samples of original dataset and then the result in the term of prediction is averaged across all these bagged trees. This leads to reduction of variance and deal with of the weaknesses of Decision Trees, which ae sensitive to variance.

However, to further improve performance, method called Random Forest (Breiman, 2001) was introduced. The main principle of Random Forest algorithm is to combines bagging with random selection of features. That means that while doing each split, not all variables in a dataset are being considered but only restricted number of them is considered each time, and the selection of given number of predictors for consideration is random. This improvement leads to decorrelated trees. It is a significant improvement of bagging method, as if there is one strong predictor so with bagging all trees would be built with top split using these strong predictors and other predictors are avoided, which causes very similar looking trees. Thus, if very similar trees had been built, so averaging of their predictions would not have lead into reduction of variance. Random Forest deals exactly with this – contribute into variance reduction and allows for exploring less obvious relationship in data.

The number of predictors to consider for each split is a matter of random forest parameter tuning, however the main recommendation is to use n/3 in the case of regression and sqrt n in the case of classification, where n is total number of predictions. Anyway, the more correlated predictors the smaller value of predictors may be found fruitful.

Another important parameter to state in Random Forest is number of trees to build. There is discrepancy in the research, and while the official documentation of random forest states it does not overfit (Breiman, 2001), so M. R. Segal (2004) found that RF may does overfit when noisy datasets are provided. Oshiro, Perez (2012) found that more trees does not necessarily lead to better performance and recommend to use 64 – 128 trees, as there were no significant improvements in performance when more than 128 were deployed. However, as noted by Plonski (2020) in his extensive research, the ideal number of trees depends on number of rows in dataset, when the more rows the more trees may be helpful.

Our setting of parameters for all tests performed:

- Number of trees: 100
- Number of Random Predictors (grid search of 3 parameters): n/3, n/3-1, n/3+1 , where n = total number of predictors in dataset
- To find optimal number of randomly selected predictors, cross validation with grid search were used.

Naturally, Random Forest method is not a holy grail, and it has its disadvantages. One of them is worse explainability then simple decision trees, and, they are much more computationally feasible then other algorithms.

# 4 Data Collection

The methodology and comparison were tested on ten randomly selected datasets that deals with any regression problem. All datasets are publicly available, and they differ in problems to solve. Importantly, they differ in descriptive statistics, which means there are difference between number of predictors, size of observations and descriptive statistics of dependent variables. These are reasons why given datasets were chosen for this study – they are publicly available, and they represent nice variance in term of task, number of predictors and number of observations.

## *4.1 Descriptive Statistics*

Table 1 [Datasets – Descriptive Statistics]

| Dataset | Dependent | N Predictors | N Observations | Mean | Median |
|---|---|---|---|---|---|
| Car Price | Price | 9 | 25639 | 18956 | 13300 |
| Energy Efficiency | Heating Load | 8 | 768 | 22.3 | 18.95 |
| Student Grade | G3 (final year grade) | 30 | 395 | 10.42 | 11 |
| Flight Price | Price | 10 | 300153 | 20890 | 7425 |
| Uber Fares | Fare Price | 7 | 200000 | 8.5 | 11.36 |
| Bank Account Opening | Age | 20 | 41188 | 40 | 38 |
| Bike Share | Shared Bikes | 12 | 8760 | 704 | 504 |
| Steel Industry Energy | Usage kWh | 9 | 35040 | 27.4 | 4.57 |
| Body Fat | Body Fat | 14 | 252 | 19 | 19 |
| Song Popularity | Song Popularity | 13 | 18835 | 53 | 56 |

# 5 Results

In this section, results of Random Forest and Single Decision Tree are compared with results of Merged Decision Trees based on our methodology, when results of 2 and 3 merged trees are merged separately. However, there was a case when merging a 3rd tree did not cause any result change and so this example was excluded (Song Popularity).

Naturally, results differ, and it is also a matter of individual datasets that influence performance of all methods. The main criteria for evaluation and comparison are Root Mean Squared Error (RMSE) as it penalizes for large error more than for smaller ones, which is not the case of other metrics, like for example Mean Absolute Error and we think that a metric that counts for large error more significantly is less biased and more reliable metric than other ones.

## *5.1 Main Insights*

Table 2 [Comparison of RMSE among all models]

| Model | 1 Tree | 2 Trees | 3 Trees | Random Forest |
|---|---|---|---|---|
| Car Price | 11633 | 11095 | 10788 | 6714 |
| Energy Efficiency | 2.66 | 1.79 | 1.73 | 0.55 |
| Student Grade | 4.86 | 3.84 | 3.8 | 3.8 |
| Flight Price | 5419 | 5335 | 5275 | 3650 |
| Uber Fares | 6.65 | 6.56 | 6.54 | 5.5 |
| Back Account Opening | 8.27 | 8.22 | 8.22 | 7.6 |
| Bike Share | 291 | 273 | 271 | 226 |
| Steel Industry Energy Consumption | 4.15 | 4.09 | 4.02 | 2.1 |
| Body Fat | 1.85 | 1.75 | 1.73 | 1.3 |
| Song Popularity | 21.5 | 21.4 | - | 16.8 |

***Car Price dataset***. Dataset of 25 639 observations and 9 predictors with mean response variable (car price) of 18 956. Single Decision Tree got relatively high RMSE 11633, however clear improvement can be seen when trees are merged in both cases. Merging of 2 best performing trees yielded to RMSE decrease by 5 % , while merging of 3 best performing decision trees decrease RMSE by 8 % compared to Single Tree and compared to 2 trees by 2.77%. In this case, merging not only two but three trees were fruitful and made sense. Regarding Random Forest, RMSE of Random Forest was 6714, meaning that Random Forest performed significantly better than all kinds of Decision Trees. However, we could decrease the difference by merging two or three trees together.

***Energy Efficiency***. Dataset of 768 observations and 8 predictors with a mean response variable (heating load) of 22.3. Single tree achieved solid RMSE 2.66, however merging 2 best performing trees decreased RMSE by 49 % compared to Single Tree, which was the most significant improvement among all other tests in the study. To merge one more tree did not yield into decrease of RMSE compared to single tree by

54 % and compared to 2 trees by 3.35 %. We can see that to merge more trees brought very significant improvement of RMSE in the term of percentage change. Regarding Random Forest, RMSE of Random Forest was 0.55 and this case was the poorest performance of (merged) trees in comparison to Random Forest in the whole study.

***Student Grade***. Dataset of 395 observations and 30 predictors with a mean response variable (final year grade) of 10.4. In term of predictors, this is dataset that contains the biggest number of predictors in the study. RMSE of Single tree was 4.86, merging of 2 best trees decreased RMSE by 27%. To merge 3rd tree does not bring any impactful advantage in this case and decrease RMSE just marginally. Regarding Random Forest, RMSE of Random Forest was 3.8. This was the case when RMSE of Single Decision Tree was higher, however by merging two and three trees we were to obtain the result as accurate as the result of Random Forest.

***Flight Price***. Dataset of 300 153 observations and 10 predictors with a mean response variable (flight price) of 20 890. In term of observations, this is the largest dataset in the study. Single tree recorded RMSE 5 419. There were some improvements when the 2nd and 3rd best trees were added into results, however decrease in RMSE was just marginal (-2% and -3%). Regarding Random Forest, RMSE of Random Forest was 3650, meaning that Random Forest performed significantly better than all kinds of Decision Trees. However, we could decrease the difference by merging two or three trees together.

***Uber Fares***. Dataset of 200 000 observations and 7 predictors with a mean response variable (fare price) of 8.5. Single tree resulted into RMSE 6.65 and merging of two best trees improved performance by decreasing RMSE by 1% only. Adding one more tree decreased RMSE by 2 % compared to Single Tree. There were not significant improvements. Regarding Random Forest, RMSE of Random Forest was 5.5, meaning that Random Forest performed significantly better than all kinds of Decision Trees. However, we could significantly decrease the difference by merging two or three trees together, when RMSE of Single Tree increased by 21% compared to RF, and RMSE of 2 and 3 merged trees increased by 19 % compared to RMSE of RF.

***Bank Account Opening***. Dataset of 41 188 observations and 20 predictors with a mean response variable (age) of 40. Single tree got RMSE 8.27 and to merge one more tree marginally improved RMSE by 1%. To merge the 3rd did not bring any additional effect and so in this case would be useless. Regarding Random Forest, RMSE of Random Forest was 7.6, which means that RF performed better however the difference in RMSE was not huge. When compared to merged trees, so RMSE increased only by 8% compared to Random Forest RMSE. This was the second best performance of Trees in comparison with Random Forest.

***Bike Share***. Dataset of 8 760 observations and 12 predictors with a mean response variable (shared bikes) of 704. Single tree resulted into RMSE of 291 and that was improved by merging another tree, when RMSE

decrease by 7% compared to single tree. To merge 3rd tree yielded only into very marginal improvement. Regarding Random Forest, RMSE of Random Forest was 226, meaning that Random Forest performed significantly better than all kinds of Decision Trees. However, we could significantly decrease the difference by merging two or three trees together, when RMSE of Single Tree increased by 29% compared to RF, however RMSE of 3 merged trees increased only by 21 % compared to RF.

***Steel Industry Energy Consumption***. Dataset of 35 040 observations and 9 predictors with a mean response variable (shared bikes) of 27.4. RMSE of single tree was 4.15 and to merge 2nd and also 3rd tree lead into some decline in RMSE but just marginal, when 2 trees decrease RMSE by 1 % and 3 trees by 3%. Regarding Random Forest, RMSE of Random Forest was 2.1, meaning that Random Forest performed significantly better than all kinds of Decision Trees and the difference was serious as RMSE of (merged) trees was almost doubled compared to RF.

***Body Fat***. Dataset of 252 observations and 14 predictors with a mean response variable (body fat) of 19. Single tree got RMSE 1.85 and tree merging recorded improvements, especially in the case of adding 2nd tree which led into decrease of RMSE by 6 % compared to Single tree. Adding the third tree caused just a minor improvement. Regarding Random Forest, RMSE of Random Forest was 2.1, meaning that Random Forest performed significantly better than all kinds of Decision Trees. However, we could decrease the difference by merging two or three trees together. RMSE of Single tree recorded increase by 42 %, however in the case of 2 and 3 merged trees it was already only by 35 % and 33 %, respectively.

***Song Popularity.*** Dataset of 18 835 observations and 13 predictors with a mean response variable (song popularity score) of 53. In this case, merging was not fruitful in any case, when adding the second tree did not cause almost any improvement and the same was the case of 3rd tree – for this reason, 3rd tree is not even included. Regarding Random Forest, RMSE of Random Forest was 16.8, meaning that Random Forest performed better than all kinds of Decision Trees. However, the differences were not crucial and even RMSE of Single tree increased just by 25 % compared to RMSE of Random Forest.

### *5.2 Decision Tree and Random Forest Results*

In general, it can be concluded that merged decision trees did lead to RMSE decrease compared to Single Decision Tree in majority of cases – in some cases more in some cases less. Also, there was a difference in improvement done by merging the 3rd tree and in some cases, it was fruitful and it some less or not at all.

This graph clearly shows in which cases was improvement in RMSE cause by merging trees significant and in which cases just minor if any. As already described above, in some dataset we could reach significant improvements by tens of precents in RMSE decline but in some other changes were lower or just very minor.

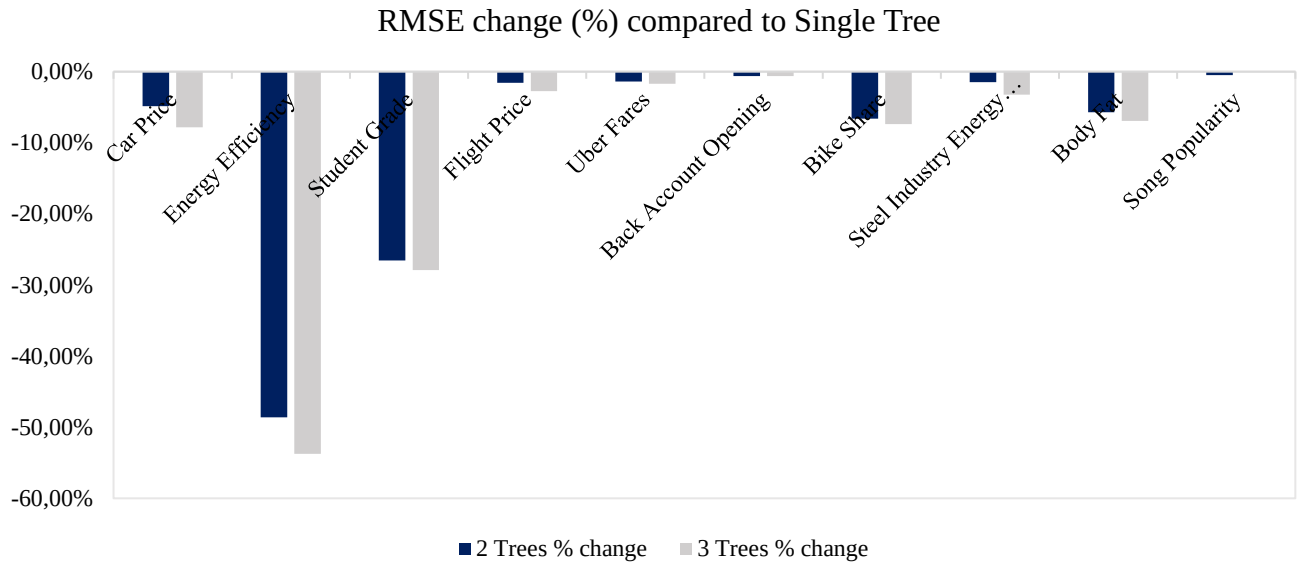



Table 3 [RMSE change (%) compared to Single Tree]

| Model | DT-Single vs. DT-2 | DT-Single vs. DT-3 |
|---|---|---|
| Car Price | -5% | -8% |
| Energy Efficiency | -49% | -54% |
| Student Grade | -27% | -28% |
| Flight Price | -2% | -3% |
| Uber Fares | -1% | -2% |
| Back Account Opening | -1% | -1% |
| Bike Share | -7% | -7% |
| Steel Industry Energy Consumption | -1% | -3% |
| Body Fat | -6% | -7% |
| Song Popularity | 0% | - |

The graph shows comparison of performance in the term of RMSE decrease between merged two and three trees. We can see that to add the third tree is usually fruitful however there is not a case when RMSE of 3 trees model would be decrease by more than 5 % compared to RMSE of 2 merged trees.

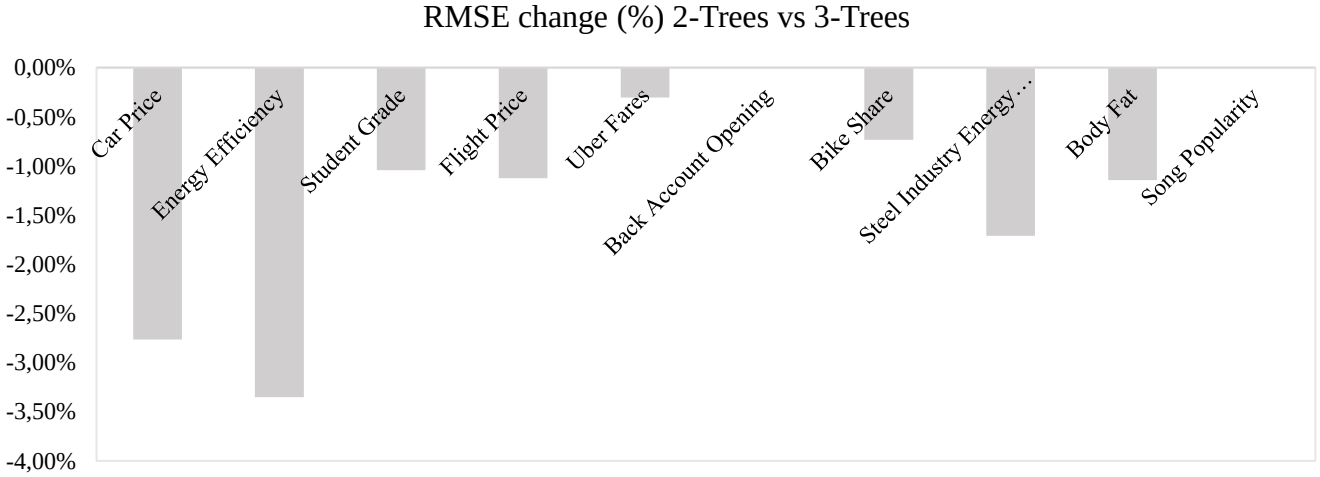


Table 4 [RMSE change (%) 2-Trees vs 3-Trees]

| Model | DT-2 vs. DT-3 |
|---|---|
| Car Price | -2.77% |
| Energy Efficiency | -3.35% |
| Student Grade | -1.04% |
| Flight Price | -1.12% |
| Uber Fares | -0.30% |
| Back Account Opening | 0.00% |
| Bike Share | -0.73% |
| Steel Industry Energy Consumption | -1.71% |
| Body Fat | -1.14% |
| Song Popularity | 0.00% |

As described in the methodology, when trees were being merged together to it was based on different weights that were related to results of individual trees in the way that it led into the smaller RMSE. This graph and table summarize distribution of weights there were given to trees in the case of 3-merged trees. It is clear that 1st and so the best performing tree usually plays the most important role, however there were cases when other trees played more important rule in the final results. For example, in datasets about flight prices, or steel industry – the third and so the least performing tree added the most important part of the final results. Similar case occurred in the case of dataset related to opening of bank account, when the second tree played the most important role.

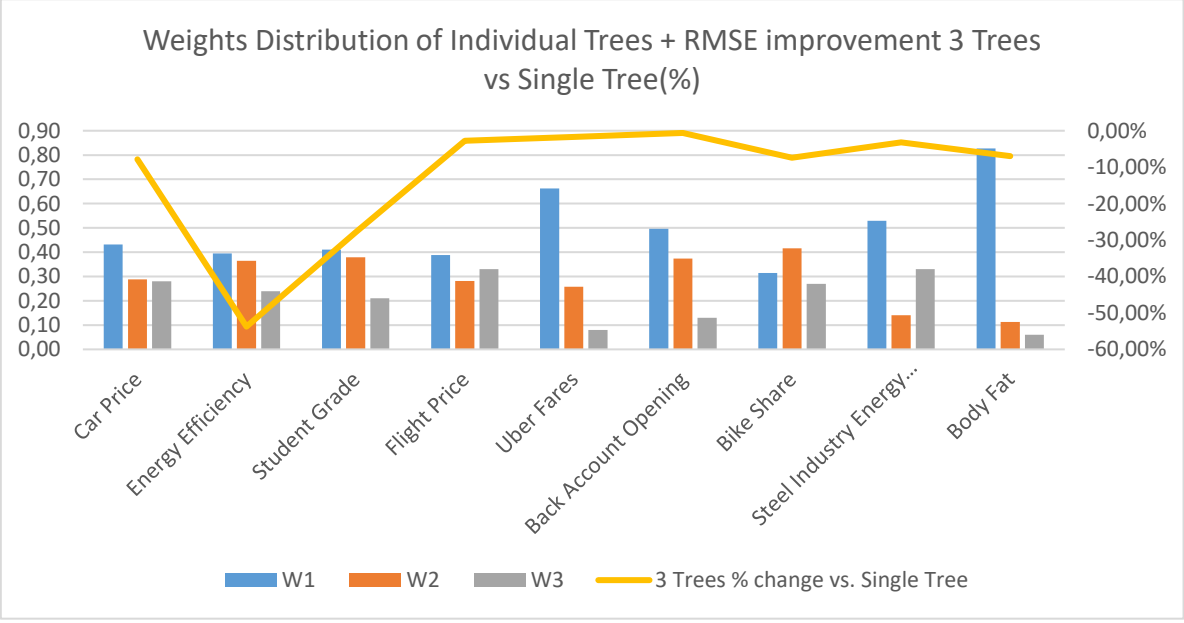



Table 5 [Weights Distribution of 3 Individual Trees + RMSE improvement 3 Trees vs. Single Tree(%)]

| | W1 | W2 | W3 | 3-T RMSE | 3 Trees % change vs. Single Tree |
|---|---|---|---|---|---|
| Car Price | 0.43 | 0.29 | 0.28 | 1078800% | -7.84% |
| Energy Efficiency | 0.40 | 0.36 | 0.24 | 1.73 | -53.76% |
| Student Grade | 0.41 | 0.38 | 0.21 | 3.8 | -27.89% |
| Flight Price | 0.39 | 0.28 | 0.33 | 5275 | -2.73% |
| Uber Fares | 0.66 | 0.26 | 0.08 | 6.54 | -1.68% |
| Back Account Opening | 0.50 | 0.37 | 0.13 | 8.22 | -0.61% |
| Bike Share | 0.31 | 0.42 | 0.27 | 271 | -7.38% |
| Steel Industry Energy Consumption | 0.53 | 0.14 | 0.33 | 4.02 | -3.23% |
| Body Fat | 0.83 | 0.11 | 0.06 | 1.73 | -6.94% |
| Song Popularity | 0.54 | 0.46 | - | - | - |

When we have a look at distributions of weights only for the cases of 2 merged trees we can see in some cases much larger differences, when the first tree played the absolutely crucial role in the final results. This applies for datasets related to Body Fat or Steel Industry or even Uber Fares, however there was a case when the second trees was more important than the first one – Bike Shares.

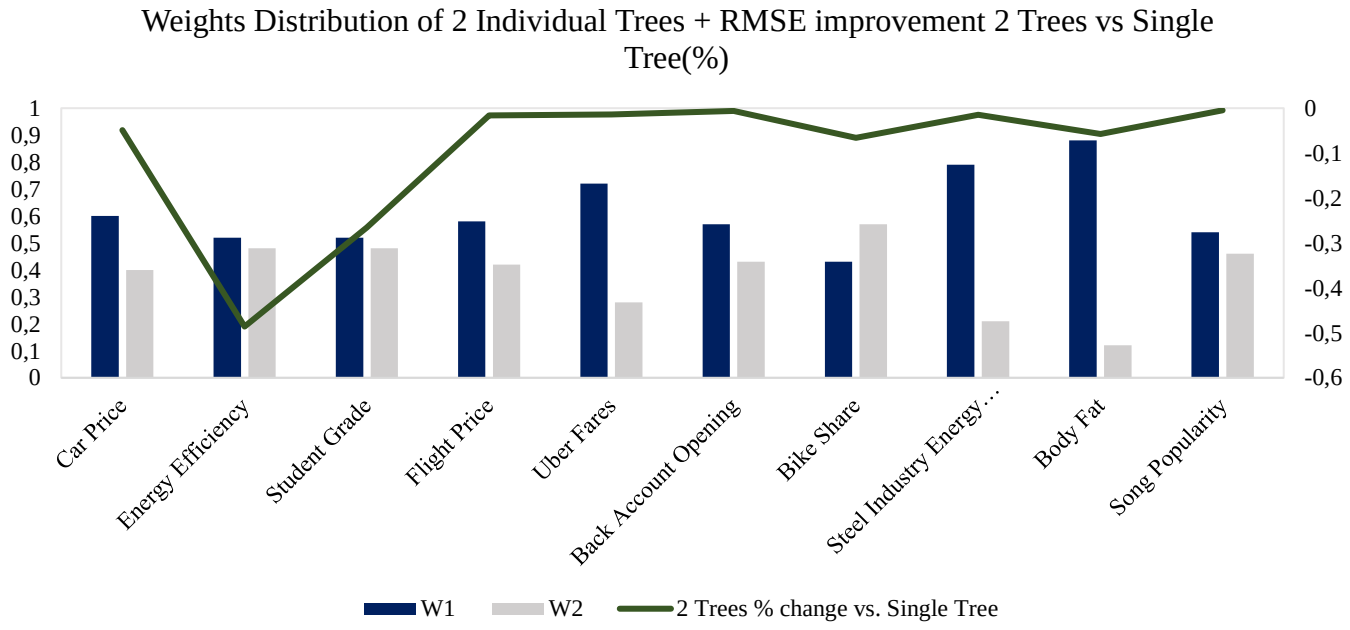


Table 6 [Weights Distribution of 2 Individual Trees + RMSE improvement 2 Trees vs Single Tree(%)]

| | W1 | W2 | 2 Trees % change vs. Single Tree |
|---|---|---|---|
| Car Price | 0.6 | 0.4 | -5% |
| Energy Efficiency | 0.52 | 0.48 | -49% |
| Student Grade | 0.52 | 0.48 | -27% |
| Flight Price | 0.58 | 0.42 | -2% |
| Uber Fares | 0.72 | 0.28 | -1% |
| Back Account Opening | 0.57 | 0.43 | -1% |
| Bike Share | 0.43 | 0.57 | -7% |
| Steel Industry Energy Consumption | 0.79 | 0.21 | -1% |
| Body Fat | 0.88 | 0.12 | -6% |
| Song Popularity | 0.54 | 0.46 | 0% |

Detailed picture of performance of Trees compared to Random Forest, that was already described above. It is clear that there were difference and, in some cases, (merged) trees were competitive and in other cases it failed (energy efficiency, steel industry). However, there were some significant performances when Decision Tree was comparable to performance of Random Forest and merged trees brought it even closer – Student Grades or Bank Account Openings dataset. While performance of 3 merged trees in Student Grades dataset was identical to Random Forest, in case of bank accounts there was RMSE increase up to 10 % in all forms of trees, compared to RMSE of Random Forest.

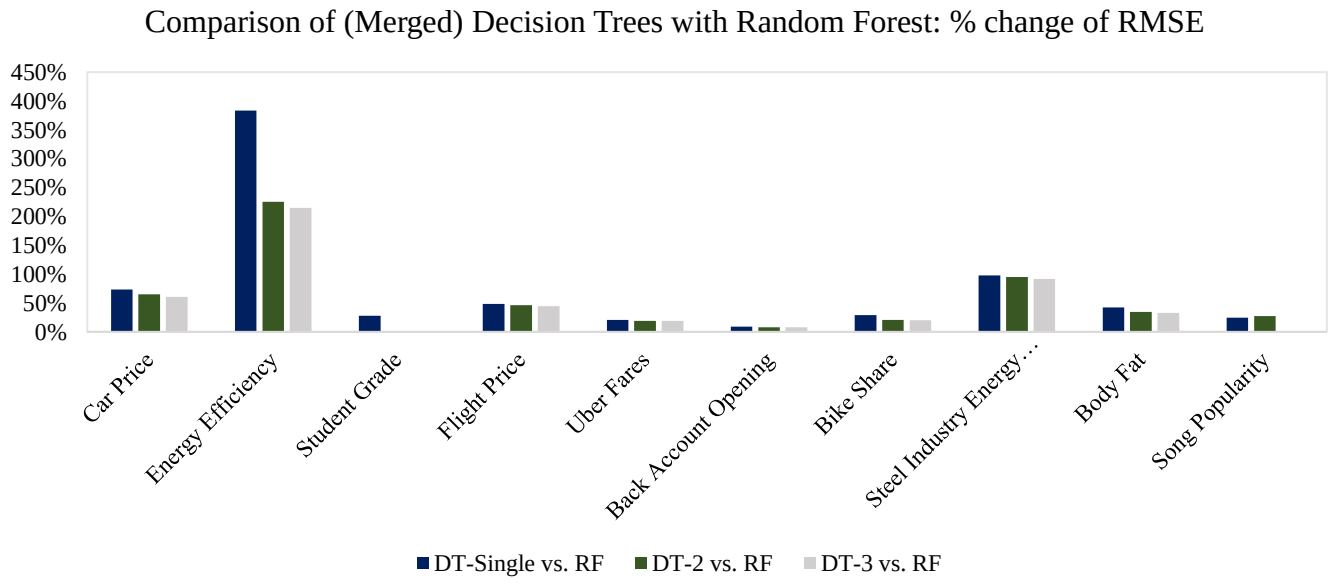
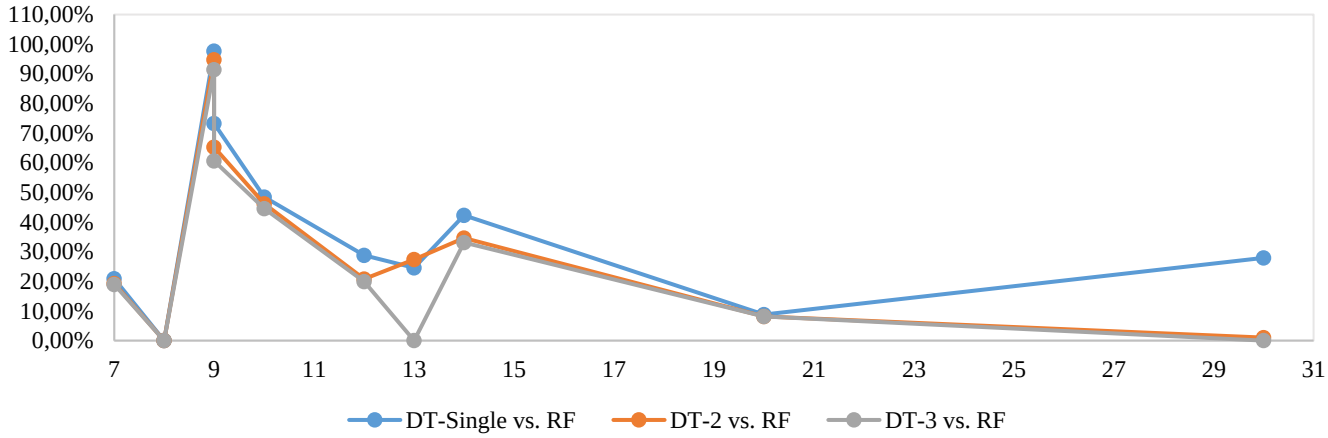


Table 7 [Comparison of (Merged) Decision Trees with Random Forest: % change of RMSE]

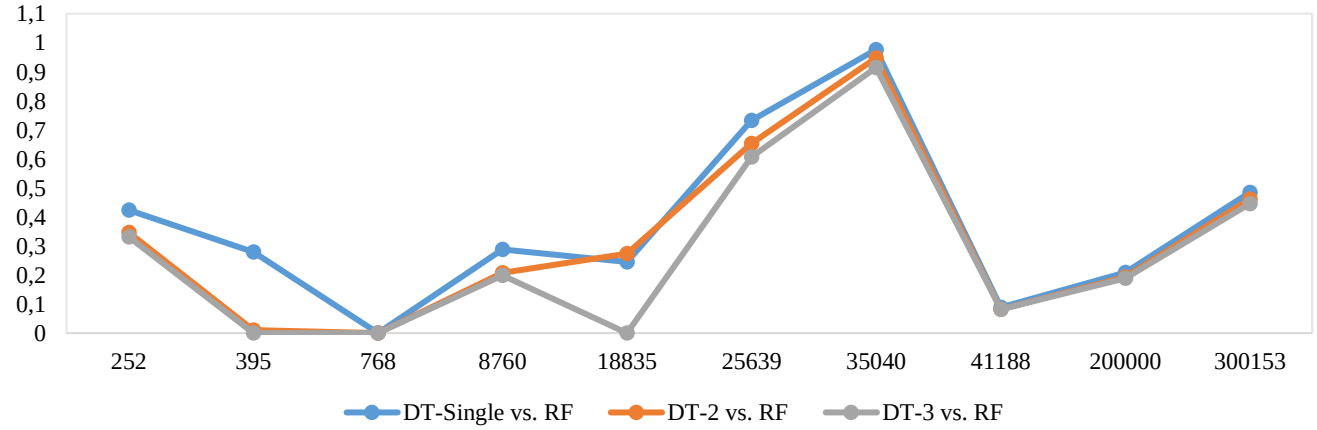
| Model | DT-Single vs. RF | DT-2 vs. RF | DT-3 vs. RF |
|---|---|---|---|
| Car Price | 73% | 65% | 61% |
| Energy Efficiency | 384% | 225% | 215% |
| Student Grade | 28% | 1% | 0% |
| Flight Price | 48% | 46% | 45% |
| Uber Fares | 21% | 19% | 19% |
| Bank Account Opening | 9% | 8% | 8% |
| Bike Share | 29% | 21% | 20% |
| Steel Industry Energy Consumption | 98% | 95% | 91% |
| Body Fat | 42% | 35% | 33% |
| Song Popularity | 25% | 27% | - |

When we take a look at differences in RMSE between RF and Trees based on number of predictors included in dataset we can see that there is decreasing trend in RMSE differences – so Trees are getting closer to RF performance – as number of predictors increase. The opposite applies to number of observation when Trees are closer to performance of RF when there are fewer observations.

RMSE % change based on N of Predictors: DTs vs Random Forest

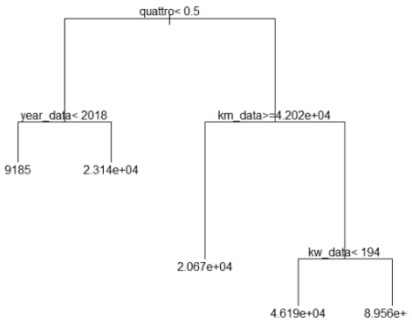


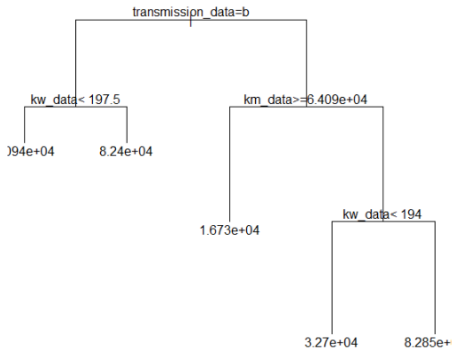

RMSE % change based on N of Observations: DTs vs Random Forest

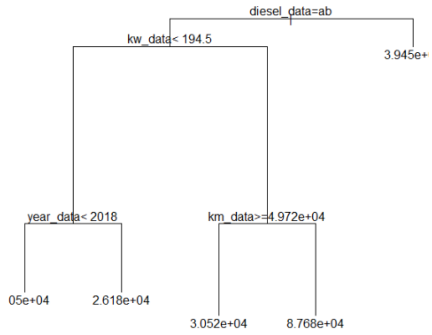


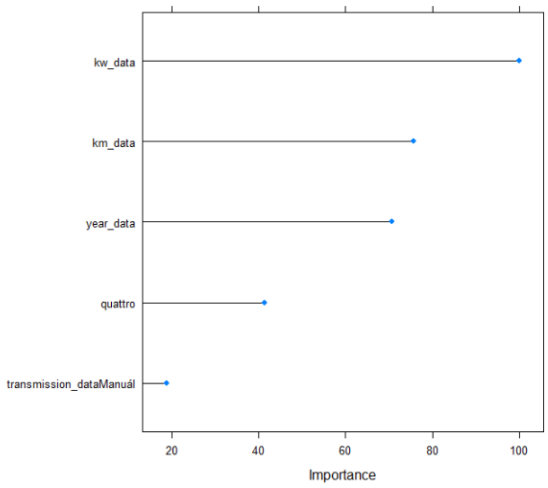
*5.3 Graphical Example of Merged Decision Trees and Random Forest*

As illustration, we have chosen merged decision trees related to Car Price dataset to compare visualizations and important variables of tree merged trees with Random Forest Variable Importance after Permutations. We can see that in the first as so the best performing tree – variable quattro was selected for the root split. This variable indicates if a car is 4x4 or not. The second tree used as its node split variable indicating type of transmission and the third tree used type of fuel. However, all three trees contain variables about mileage (km_data) and power (kw_data). When we compare Variable Importance of Random Forest, we see that the most important variables are variables about mileage and power – so the ones that are included in all merged trees, however not as a root splits. However, variables about 4x4 and transmission are still included in top 5 the most important variables for Random Forest. Thus, we can indicate that selection of important variables in merged decision trees and random forest tend to be similar.



# 6 Limitations

First – only ten datasets were used for testing purposes. It may be useful to consider more datasets that would differ in more ways and compare results again. Secondly, the performance of Random Forest depends on tuning parameters and there is the possibility that the best tuning parameter was not found in all cases and so it biased the real performance of RF that could have been achieved. However, the same implies to decision tree as they were built in a way to not be deep and complex.

# 7 Discussion and Conclusion

It is clear and expected that Random Forest performs better than Decision Trees, however, there are cases and datasets when Decision Trees can be really useful and not only, they can achieve good accuracy but mainly they keep their interpretability and simpleness for audience. In our datasets, decision tree performed the best in case of dataset (student grade) with not many observations (395) and relatively many predictors (30). Decision Tree performed only slightly worse than Random Forest and merged decision trees got the same performance as Random Forest. However, there was also a case when Decision Trees did not do bad when used for large dataset of 41 188 observations and 20 predictors. The increase of RMSE for both merged and single decision trees was in all cases below 10 %, which is competitive results. Thus, it may be concluded that as stated by Rudin (2019), white-box models do not have to necessarily lead to worse performance than black-box models. It is necessary to note that all decision trees were constructed in the way to remain their interpretability and so tuning parameters were given in the way to do not allow building very complex trees with many branches, which was always also inspected visually. We are sure that if we would not take care about remaining interpretability, so performance of decision tree could be even better and close to Random Forest.

Regarding our new proposed method of merged decision trees, we can conclude that in most cases it led improvement of performance compared to just single decision tree. This is due to wider spread or variability that can be considered in the final output when outputs of more and different trees, which includes more variables and rules and so the discrepancy in observed data can be described and learnt somehow better.

Naturally, in the most cases the most important part of the result consisted of the best performing tree (the highest weight in the final prediction), however the second best-performing tree usually played important part and there were high weights given to outputs of second trees. There was even a case when the prediction of second best-performing tree did better and so it got higher weight than the first best-performer. However, the third trees played its part on the results too and it was very rare if the difference between weight of $2^{nd}$ and $3^{rd}$ tree was significant. There were occurrences when weight of the third tree was the same as the weight of $2^{nd}$ tree or rarely even higher. Thus, it is worth to try to build more trees with

different setting – in our case with different starting points and to try their performance, no to rely only on the best tree because it may be fruitful.

The importance of interpretability and transparency of machine learning and artificial intelligence solutions is increasing, and it is hot topic nowadays. Not only because people tend to trust systems, they can at least a bit understand but also because many important decisions are being made based on outputs of algorithms and no one can be sure if they are always trustworthy if there is no interpretation available.

The development of machine learning and artificial intelligence solutions is huge nowadays. These systems play very important roles in many aspects of people's lives, businesses and society.
Serious decision in all areas are based on ML & AI solutions. From finance to healthcare and justice. However, people do not often understand these systems and it may undermine their trust in them (Ray, 2020).
Thus, interpretable machine learning has become unavoidable. However, this is not only willingness, but it also comes from regulations. Regulators understand how impactful ML & AI algorithms are in nowadays society as well as they understand their drawback and risk that this decision may be biased, unreliable or discriminatory.
There are many interpretable white-box models for centuries that can be a little bit in shadow of more recent methods, so called black-box models. This can be caused by general believe that simpler and older white-box models, such as decision trees or linear models, cannot perform well and so more advanced techniques that lack interpretability should have been deployed.
However, as proved by Rudin (2019) and partially also by our research, this does not always need to be a case and there are cases when white-box models perform almost as good as black-box. In this situation, the final users of models must make a decision if they want to sacrifice some accuracy in order to have a transparent and interpretable model. However, as noted by Doshi-Velez and Kim (2017), explanations are not always necessary - especially if there are no serious consequences of wrong predictions or when we can trust the system because it was already well-studied and applied in real conditions.
To deploy the black-box model does not mean that there are no other ways how to obtain some interpretations, however methods such as LIME, are just other models that are generalized for any model and are still based on other assumptions, thus interpretations do not come directly from the background of the model.
Many researchers focus on the development of interpretable models, some of them combine more models together (Wang et al, 2018) or some of them focus on feature engineering in order to provide interpretable models with new unknown information (Broniatowski, D., A., 2021). We proposed a method that is based on decision trees, that are built with a different root, validated and their outputs merged together with given weights, which lead to the highest accuracy. It was shown that this methodology can improve the performance of a single decision tree - in some cases very significantly, in others less. We showed the

importance of other trees than only the best-performing one in the final output, as it is clear that worse-performing trees still count for high weights in the final output.
Importantly, we showed that Random Forest is a more accurate predictor in almost all cases, however, there are situations when decision trees perform almost on the same level and after improvements from merged trees it can get in some datasets really close to Random Forest's performance. Thus, it is important to keep white-box models in the mind and to not rely only on black-box models.

# 8 Future Research

Future research can go beyond the regression problems and consider classification. One example can be the rule of the major vote. Several trees with different starting points can be trained and classification can be decided upon the major vote of these trees. However, this thesis aimed only on regression problem.

# Appendix

## Datasets

| Dataset Name | Source | Link |
|---|---|---|
| Car Price | own web-scrape | own web-scrape |
| Energy Efficiency | Kaggle | https://www.kaggle.com/datasets/ujjwalchowdhury/energy-efficiency-data-set |
| Student Grade | Kaggle | https://www.kaggle.com/datasets/dipam7/student-grade-prediction |
| Flight Price | Kaggle | https://www.kaggle.com/datasets/shubhambathwal/flight-price-prediction |
| Uber Fares | Kaggle | https://www.kaggle.com/datasets/yasserh/uber-fares-dataset |
| Back Account Opening | Kaggle | https://www.kaggle.com/datasets/prakharrathi25/banking-dataset-marketing-targets |
| Bike Share | ICS UCI | https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset |
| Steel Industry Energy Consumption | Kaggle | https://www.kaggle.com/datasets/csafrit2/steel-industry-energy-consumption |
| Body Fat | Kaggle | https://www.kaggle.com/datasets/fedesoriano/body-fat-prediction-dataset |
| Song Popularity | Kaggle | https://www.kaggle.com/datasets/yasserh/song-popularity-dataset |

***Car Price***. Dataset containing data about second-hand car market in Slovakia (web-scrapper was built personally by myself). The aim is to predict used car price based on car characteristics, such as mileage, age, brand etc.

***Energy Efficiency***. Dataset contains information about different parameters of building and their energy consumption. The aim is to predict heating load in buildings based on their characteristics, such as glazing area, orientation etc.

***Student Grade***. This data examines student success at two high schools based on student's social, demographics and school qualities. The aim is to predict final year grade without knowledge of grades from 1st and 2nd period, which are strong correlates.

***Flight Price***. Data from online flight ticket seller containing information related to flights, airports and airlines, such as duration, stops, time or class. The aim is to predict flight ticket prices for different airlines and various flights settings.

***Uber Fares***. Dataset consists of characteristics of various Uber fares and their prices. The aim is to predict price of fare based on these characteristics, such as location, number of passengers or time.

***Bank Account Opening***. Dataset about different customers of a bank containing not only personal characteristics but also information about their relationship with bank and their products. The target is to predict age of customer who is about to open a bank account.

***Bike Share***. The dataset comprises the hourly and daily count of rental bikes in bike share system, together with the accompanying weather and seasonal statistics. The goal was to predict number of shared bikes.

***Steel Industry Energy Consumption***. Data collected from a company that produces several types of coils, steel and iron plates. The aim was to predict energy consumption in kWh based on characteristics related to company machines, different time period and so on.

***Body Fat***. Dataset containing personal body characteristics of different people. The aim was to predict their percentage share of body fat.

***Song Popularity***. Dataset consists of different songs and their unique features, such as duration but mainly indicating style of a song and type of music included. Based on these features, popularity of a song is being predicted.

# References


Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE access*, *6*, 52138-52160.

Apley, D. W., & Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *82*(4), 1059-1086.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*(2), 123-140.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Broniatowski, D. A. (2021). Psychological foundations of explainability and interpretability in artificial intelligence. *NIST Tech. Rep*, 1-56.

Carvalho, D. V., Pereira, E. M., & Cardoso, J. S. (2019). Machine learning interpretability: A survey on methods and metrics. *Electronics*, *8*(8), 832.

Chen, C., Lin, K., Rudin, C., Shaposhnik, Y., Wang, S., & Wang, T. (2018). An interpretable model with globally consistent explanations for credit risk. A*rXiv*, 1-10.

Donnelly, C., & Embrechts, P. (2010). The devil is in the tails: actuarial mathematics and the subprime mortgage crisis. *ASTIN Bulletin: The Journal of the IAA*, *40*(1), 1-33.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning.(2017). *ArXiv*, *27*, 1-13.

Fisher, A., Rudin, C., & Dominici, F. (2019). All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *J. Mach. Learn. Res.*, *20*(177), 1-81.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, 1189-1232.

Garcia, I. D. C. G., Sengupta, D., Lorenzo, M. M. G., & Nowe, A. (2016, September). Grey-Box Model: An ensemble approach for addressing semi-supervised classification problems. In *25th Belgian-Dutch Conference on Machine Learning* (pp. 1-3).

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). Resampling methods. In *An introduction to statistical learning* (pp. 175-201). Springer, New York, NY.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112, p. 18). New York: springer.

Killick, R., Fearnhead, P., & Eckley, I. A. (2012). Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, *107*(500), 1590-1598.

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling* (Vol. 26, p. 13). New York: Springer.

Mingers, J. (1989). An empirical comparison of pruning methods for decision tree induction. *Machine learning*, *4*(2), 227-243.

Molnar, C. (2019). *Interpretable Machine Learning*. Github. Retrieved July 20, 2022, from https://christophm.github.io/interpretable-ml-book/

Oshiro, T. M., Perez, P. S., & Baranauskas, J. A. (2012, July). How many trees in a random forest?. In *International workshop on machine learning and data mining in pattern recognition* (pp. 154-168). Springer, Berlin, Heidelberg.

Pintelas, E., Livieris, I. E., & Pintelas, P. (2020). A grey-box ensemble model exploiting black-box accuracy and white-box intrinsic interpretability. *Algorithms*, *13*(1), 17.

Płoński, P. (2020, June 30). *How many trees in the Random Forest?* MLJAR. Retrieved June 14, 2022, from https://mljar.com/blog/how-many-trees-in-random-forest/

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81-106.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " Why should I trust you?" Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135-1144.

Segal, M. R. (2004). Machine learning benchmarks and random forest regression.

Ustun, B., Traca, S., & Rudin, C. (2013). Supersparse linear integer models for interpretable classification. A*rXiv*, *1-37*.

Varshney, K. R., & Alemzadeh, H. (2017). On the safety of machine learning: Cyber-physical systems, decision sciences, and data products. *Big data*, *5*(3), 246-255.

Wachter, S., Mittelstadt, B., & Floridi, L. (2017). Why a right to explanation of automated decision-making does not exist in the general data protection regulation. *International Data Privacy Law*, *7*(2), 76-99.

Wang, J., Oh, J., Wang, H., & Wiens, J. (2018). Learning credible models. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2417-2426).

Wei, N., Yin, L., Li, C., Li, C., Chan, C., & Zeng, F. (2021). Forecasting the daily natural gas consumption with an accurate white-box model. *Energy*, *232*, 121036.

Wexler, R. (2017). When a computer program keeps you in jail. *The New York Times*, *13*.

Zeng, J., Ustun, B., & Rudin, C. (2017). Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *180*(3), 689-722.