Quantifying Model Complexity via Functional Decomposition for Better Post-Hoc Interpretability

Christoph Molnar, Giuseppe Casalicchio, and Bernd Bischl

Department of Statistics, LMU Munich,
Ludwigstr. 33, 80539 Munich, Germany
christoph.molnar@stat.uni-muenchen.de

Abstract. Post-hoc model-agnostic interpretation methods such as partial dependence plots can be employed to interpret complex machine learning models. While these interpretation methods can be applied regardless of model complexity, they can produce misleading and verbose results if the model is too complex, especially w.r.t. feature interactions. To quantify the complexity of arbitrary machine learning models, we propose model-agnostic complexity measures based on functional decomposition: number of features used, interaction strength and main effect complexity. We show that post-hoc interpretation of models that minimize the three measures is more reliable and compact. Furthermore, we demonstrate the application of these measures in a multi-objective optimization approach which simultaneously minimizes loss and complexity.

Keywords: Model Complexity · Interpretable Machine Learning · Explainable AI · Accumulated Local Effects · Multi-Objective Optimization

1 Introduction

Machine learning models are optimized for predictive performance, but it is often required to understand models, e.g., to debug them, gain trust in the predictions, or satisfy regulatory requirements. Many post-hoc interpretation methods either quantify effects of features on predictions, compute feature importances, or explain individual predictions, see [17, 24] for more comprehensive overviews. While model-agnostic post-hoc interpretation methods can be applied regardless of model complexity [30], their reliability and compactness deteriorates when models use a high number of features, have strong feature interactions and complex feature main effects. Therefore, model complexity and interpretability are deeply intertwined and reducing complexity can help to make model interpretation more reliable and compact. Model-agnostic complexity measures are needed to strike a balance between interpretability and predictive performance [4, 31].

Contributions. We propose and implement three model-agnostic measures of machine learning model complexity which are related to post-hoc interpretability. To our best knowledge, these are the first model-agnostic measures that describe the global interaction strength, complexity of main effects and number

of features. We apply the measures to different datasets and machine learning models. We argue that minimizing these three measures improves the reliability and compactness of post-hoc interpretation methods. Finally, we illustrate the use of our proposed measures in multi-objective optimization.

2 Related Work and Background

In this section, we introduce the notation, review related work, and describe the functional decomposition on which we base the proposed complexity measures.

Notation: We consider machine learning prediction functions $f : \mathbb{R}^p \mapsto \mathbb{R}$, where $f(x)$ is a prediction (e.g., regression output or a classification score). For the decomposition of f, we write $f_S : \mathbb{R}^{|S|} \mapsto \mathbb{R}$, $S \subseteq \{1, \ldots, p\}$, to denote a function that maps a vector $x_S \in \mathbb{R}^{|S|}$ with a subset of features to a marginal prediction. If subset S contains a single feature j, we write f_j. We refer to the training data of the machine learning model with the tuples $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^n$ and refer to the value of the j-th feature from the i-th instance as $x_j^{(i)}$. We write X_j to refer to the j-th feature as a random variable.

Complexity and Interpretability Measures: In the literature, model complexity and (lack of) model interpretability are often equated. Many complexity measures are model-specific, i.e., only models of the same class can be compared (e.g., decision trees). Model size is often used as a measure for interpretability (e.g., number of decision rules, tree depth, number of non-zero coefficients) [3, 16, 20, 22, 31–34]. Akaikes Information Criterion (AIC) and the Bayesian Information Criterion (BIC) are more widely applicable measures for the trade-off between goodness of fit and degrees of freedom. In [26], the authors propose model-agnostic measures of model stability. In [27], the authors propose explanation fidelity and stability of local explanation models. Further approaches measure interpretability based on experimental studies with humans, e.g., whether humans can predict the outcome of the model [8, 13, 20, 28, 35].

Functional Decomposition: Any high-dimensional prediction function can be decomposed into a sum of components with increasing dimensionality:

$$f(x) = \overbrace{f_0}^{\text{Intercept}} + \overbrace{\sum_{j=1}^{p} f_j(x_j)}^{\text{1st order effects}} + \overbrace{\sum_{j<k}^{p} f_{jk}(x_j, x_k)}^{\text{2nd order effects}} + \ldots + \overbrace{f_{1,\ldots,p}(x_1, \ldots, x_p)}^{\text{p-th order effect}} \quad (1)$$

This decomposition is only unique with additional constraints regarding the components. Accumulated Local Effects (ALE) were proposed in [1] as a tool for visualizing feature effects (e.g., Figure 1) and as unique decomposition of the prediction function with components $f_S = f_{S,ALE}$. The ALE decomposition is unique under an orthogonality-like property described in [1].

The ALE main effect $f_{j,ALE}$ of a feature $x_j, j \in \{1, \ldots, p\}$ for a prediction function f is defined as

$$f_{j,ALE}(x_j) = \int_{z_{0,j}}^{x_j} \mathbb{E}\left[\frac{\partial f(X_1, \ldots, X_p)}{\partial X_j} \middle| X_j = z_j\right] dz_j - c_j \quad (2)$$

Here, $z_{0,j}$ is a lower bound of X_j (usually the minimum of x_j) and the expectation $\mathbb{E}$ is computed conditional on the value for x_j and over the marginal distribution of all other features. The constant c_j is chosen so that the mean of $f_{j,ALE}(x_j)$ with respect to the marginal distribution of X_j is zero, so that the ALE components sum to the full prediction function. By integrating the expected derivative of f with respect to X_j the effect of x_j on the prediction function f is isolated from the effects of all other features. ALE main effects are estimated with finite differences, i.e., access to the gradient of a prediction function is not required (see [1]). We base our proposed measures on the ALE decomposition, because ALE are computationally cheap (worst case $O(n)$ per main effect), they can be computed sequentially instead of simultaneously, they do not require knowledge of the joint distribution, and several software implementations exist [2, 25].

3 Functional Complexity

In this section, we motivate complexity measures based on functional decomposition. Based on Equation 1, we decompose the prediction function into a constant (estimated as $f_0 = \frac{1}{n}\sum_{i=1}^{n} f(x^{(i)})$), main effects (estimated by ALE), and a remainder term containing interactions (i.e., the difference between the full model and constant + main effects).

$$f(x) = \underbrace{f_0 + \sum_{j=1}^{p} \overbrace{f_{j,ALE}(x_j)}^{\text{MEC: How complex?}} + \overbrace{IA(x)}^{\text{IAS: Interaction strength?}}}_{\text{NF: How many features were used?}} \tag{3}$$

This arrangement of components emphasizes a decomposition of the prediction function into a main effect model and an interaction remainder. We can analyze how well the main effect model itself approximates f by looking at the magnitude of the interaction measure IAS. The average main effect complexity (MEC) captures how many parameters are needed to describe the one-dimensional main effects on average. The number of features used (NF) describes how many features were used in the full prediction function.

3.1 Number of Features (NF)

We propose an approach based on feature permutation to determine how many features are used by a model. We regard features as "used" when changing a feature changes the prediction. If available, the model-specific number of features is preferable. The model-agnostic version is useful when the prediction function is only accessible via API or when the machine learning pipeline is complex.

The proposed procedure is formally described in Algorithm 1. To estimate whether the j-th feature was used, we sample instances from data $\mathcal{D}$, replace their j-th feature values with random values from the distribution of X_j (e.g., by

Algorithm 1: Number of Features Used (NF)

Input: Number of samples M, data $\mathcal{D}$
NF = 0
for $j \in 1, \ldots, p$ **do**
  Draw M instances $\{x^{(m)}\}_{m=1}^{M}$ from dataset $\mathcal{D}$
  Create $\{x^{(m)*}\}_{m=1}^{M}$ as a copy of $\{x^{(m)}\}_{m=1}^{M}$
  for $m \in 1, \ldots, M$ **do**
    Sample $x_j^{(new)}$ from $\{x_j^{(i)}\}_{i=1}^{n}$ with the constraint that $x_j^{(new)} \neq x_j^{(m)}$
    Set $x_j^{(m)*} = x_j^{(new)}$
  if $\hat{f}(x^{(m)*}) \neq \hat{f}(x^{(m)})$ *for any* $m \in \{1, \ldots, M\}$ **then** $NF = NF + 1$.
return NF

sampling x_j from other instances from $\mathcal{D}$), and observe whether the predictions change. If the prediction of any sample changes, the feature was used.

We tested the NF heuristic with the Boston Housing data. We trained decision trees (CART) with maximum depths $\in \{1, 2, 10\}$ leading to 1, 2 and 4 features used and an L1-regularized linear model with penalty $\lambda \in \{10, 5, 2, 1, 0.1, 0.001\}$ leading to 0, 2, 3, 4, 11 and 13 features used. For each model, we estimated NF with sample sizes $M \in \{10, 50, 500\}$ and repeated each estimation 100 times. For the elastic net models, NF was always equal to the number of non-zero weights. For CART, the mean absolute differences between NF and number of features used in the trees were 0.300 ($M = 10$), 0.020 ($M = 50$) and 0.000 ($M = 500$).

3.2 Interaction Strength (IAS)

Interactions between features mean that the prediction cannot be expressed as a sum of independent feature effects, but the effect of a feature depends on values of other features [24]. We propose to measure interaction strength as the scaled approximation error between the ALE main effect model and the prediction function f. Based on the ALE decomposition, the ALE main effect model is defined as the sum of first order ALE effects:

$$f_{ALE1st}(x) = f_0 + f_{1,ALE}(x_1) + \ldots + f_{p,ALE}(x_p)$$

We define interaction strength as the approximation error measured with loss L:

$$IAS = \frac{\mathbb{E}(L(f, f_{ALE1st}))}{\mathbb{E}(L(f, f_0))} \geq 0 \tag{4}$$

Here, f_0 is the mean of the predictions and can be interpreted as the functional decomposition where all feature effects are set to zero. IAS with the $L2$ loss equals 1 minus the R-squared measure, where the true targets y_i are replaced with $f(x^{(i)})$.

$$IAS = \frac{\sum_{i=1}^{n}(f(x^{(i)}) - f_{ALE1st}(x^{(i)}))^2}{\sum_{i=1}^{n}(f(x^{(i)}) - f_0)^2} = 1 - R^2$$

If $IAS = 0$, then $L(f, f_{ALE1st}) = 0$, which means that the first order ALE model perfectly approximates f and the model has no interactions.

3.3 Main Effect Complexity (MEC)

To determine the average shape complexity of ALE main effects $f_{j,ALE}$, we propose the main effect complexity (MEC) measure. For a single ALE main effect, we define MEC_j as the number of parameters needed to approximate the curve with piece-wise linear models. For the entire model, MEC is the average MEC_j over all main effects, weighted with their variance. Figure 1 shows an ALE plot (= main effect) and its approximation with two linear segments.

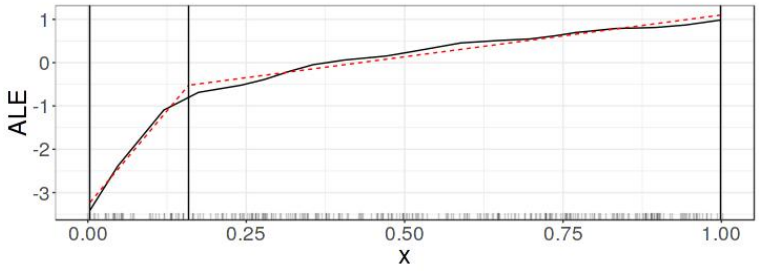


Fig. 1. ALE curve (solid line) approximated by two linear segments (dotted line).

We use piece-wise linear regression to approximate the ALE curve. Within the segments, linear models are estimated with ordinary least squares. The breakpoints that define the segments are found by greedy and exhaustive search along the interval boundaries of the ALE curve. Greedy here means that we first optimize the first breakpoint, then the second breakpoint with the first breakpoint fixed and so on. We measure the degrees of freedom as the number of non-zero coefficients for intercepts and slopes of the linear models. The approximation allows some error, e.g., an almost linear main effect may have $\text{MEC}_j = 1$, even if dozens of parameters would be needed to describe it perfectly. The approximation quality is measured with R-squared (R^2), i.e., the proportion of variance of $f_{j,ALE}$ that is explained by the approximation with linear segments. An approximation has to reach an $R^2 \geq 1 - \epsilon$, where ϵ is the user defined maximum approximation error. We also introduced parameter max_{seg}, the maximum number of segments. In the case that an approximation cannot reach an $R^2 \geq 1 - \epsilon$ with a given max_{seg}, MEC_j is computed with the maximum number of segments. The selected maximum approximation error ϵ should be small, but not too small. We found ϵ between 0.01 and 0.1 visually meaningful (i.e. a subjectively good approximation) and used $\epsilon = 0.05$ throughout the paper. We apply a post-processing step that greedily sets slopes of the linear segments to zero, as long as $R^2 \in \{1 - \epsilon, 1\}$. The post-processing potentially decreases the MEC_j,

especially for models with constant segments like decision trees. MEC_j is averaged over all features to obtain the global main effect complexity. Each MEC_j is weighted with the variance of the corresponding ALE main effect to give more weight to features that contribute more to the prediction. Algorithm 2 describes the MEC computation in detail.

Algorithm 2: Main Effect Complexity (MEC).

Input: Model f, approximation error ϵ, max. segments max_{seg}, data $\mathcal{D}$

Define $R^2(g_j, f_{j,ALE}) := \sum_{i=1}^{n}(g_j(x_j^{(i)}) - f_{j,ALE}(x_j^{(i)}))^2 / \sum_{i=1}^{n}(f_{j,ALE}(x_j^{(i)}))^2$

for $j \in \{1, \ldots, p\}$ **do**

  Estimate $f_{j,ALE}$

  `// Approximate ALE with linear model`

  Fit $g_j(x_j) = \beta_0 + \beta_1 x_j$ predicting $f_{j,ALE}(x_j^{(i)})$ from $x_j^{(i)}$, $i \in 1, \ldots, n$

  Set $K = 1$

  `// Increase nr. of segments until approximation is good enough`

  while $K < max_{seg}$ *AND* $R^2(g_j, f_{j,ALE}) < (1 - \epsilon)$ **do**

    `// Find intervals` Z_k `through exhaustive search along ALE curve breakpoints`

    `// For categorical feature, set slopes` $\beta_{1,k}$ `to zero`

    $g_j(x_j) = \sum_{k=1}^{K+1} \mathbb{I}_{x_j \in Z_k} \cdot (\beta_{0,k} + \beta_{1,k} x_j)$

    Set $K = K + 1$

  Greedily set slopes to zero while $R^2 > 1 - \epsilon$

  `// Sum of non-zero coefficients minus first intercept`

  $MEC_j = K + \sum_{k=1}^{K} \mathbb{I}_{\beta_{1,k} > 0} - 1$

  $V_j = \frac{1}{n} \sum_{i=1}^{n}(f_{j,ALE}(x^{(i)}))^2$

return $MEC = \frac{1}{\sum_{j=1}^{p} V_j} \sum_{j=1}^{p} V_j \cdot MEC_j$

4 Application of Complexity Measures

In the following experiment, we train various machine learning models on different prediction tasks and compute the model complexities. The goal is to analyze how the complexity measures behave across different datasets and models. The dataset are: Bike Rentals [10] (n=731; 3 numerical, 6 categorical features), Boston Housing (n=506; 12 numerical, 1 categorical features), (downsampled) Superconductivity [18] (n=2000; 81 numerical, 0 categorical features) and Abalone [9] (n=4177; 7 numerical, 1 categorical features).

Table 1 shows performance and complexity of the models. As desired, the main effect complexity for linear models is 1 (except when categorical features with 2+ categories are present as in the bike data), and higher for more flexible methods like random forests. The interaction strength (IAS) is zero for additive models (boosted GAM, (regularized) linear models). Across datasets we observe

	bike				Boston Housing				superconductivity				abalone			
learner	MSE	MEC	IAS	NF	MSE	MEC	IAS	NF	MSE	MEC	IAS	NF	MSE	MEC	IAS	NF
cart	905974	1.2	0.07	6	26.6	1.9	0.12	4	329.0	1.0	0.27	8	5.9	2.8	0.09	3
cart2	1307619	1.0	0.01	2	34.6	1.7	0.02	2	431.4	1.0	0.27	3	6.6	3.0	0.02	1
cvglmnet	686320	1.2	0.00	9	27.7	1.0	0.00	9	349.3	1.0	0.00	45	5.2	1.0	0.00	7
gamboost	531245	1.6	0.00	8	16.5	2.5	0.00	10	362.1	2.1	0.00	17	5.3	1.1	0.00	4
ksvm	403762	1.6	0.04	8	16.4	1.7	0.09	13	268.5	2.2	0.22	81	4.6	1.0	0.11	8
lm	636956	1.5	0.00	9	23.0	1.0	0.00	13	330.2	1.0	0.00	81	4.9	1.0	0.00	8
rf	460362	1.8	0.06	9	12.0	2.4	0.11	13	180.8	2.9	0.21	81	4.6	1.7	0.29	8

Table 1. Model performance and complexity on 4 regression tasks for various learners: linear models (lm), cross-validated regularized linear models (cvglmnet), kernel support vector machine (ksvm), random forest (rf), gradient boosted generalized additive model (gamboost), decision tree (cart) and decision tree with depth 2 (cart2).

that the underlying complexity measured as the range of MEC and IAS across the models varies. The bike dataset seems to be adequately described by only additive effects, since even random forests, which often model strong interactions show low interaction strength here. In contrast, the superconductivity dataset is better explained by models with more interactions. For the abalone dataset there are two models with low MSE: the support vector machine and the random forest. We might prefer the SVM, since main effects can be described with single numbers ($MEC = 1$) and interaction strength is low.

5 Improving Post-hoc Interpretation

Minimizing the number of features (NF), the interaction strength (IAS), and the main effect complexity (MEC) improves reliability and compactness of post-hoc interpretation methods such as partial dependence plots, ALE plots, feature importance, interaction effects and local surrogate models.

Fewer features, more compact interpretations. Minimizing the number of features improves the readability of post-hoc analysis results. The computational complexity and output size of most interpretation methods scales with $O(\text{NF})$, like feature effect plots [1, 14] or feature importance [6, 11]. As demonstrated in Table 2, a model with fewer features has a more compact representation. If additionally $IAS = 0$, the ALE main effects fully characterize the prediction function. Interpretation methods that analyze 2-way feature interactions scale with $O(\text{NF}^2)$. A complete functional decomposition requires to estimate $\sum_{k=1}^{NF} \binom{NF}{k}$ components which has a computational complexity of $O(2^{NF})$.

Less interaction, more reliable feature effects. Feature effect plots such as partial dependence plots and ALE plots visualize the marginal relationship between a feature and the prediction. The estimated effects are averages across instances. The effects can vary greatly for individual instances and even have opposite directions when the model includes feature interactions.

In the following simulation, we trained three models with different capabilities of modeling interactions between features: a linear regression model, a support vector machine (radial basis kernel, C=0.05), and gradient boosted trees. We

simulated 500 data points with 4 features and a continuous target based on [15]. Figure 2 shows an increasing interaction strength depending on the model used. More interaction means that the feature effect curves become a less reliable summary of the model behavior.

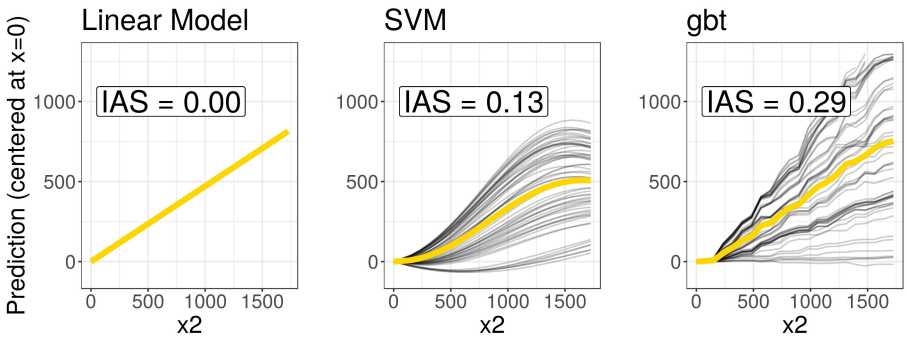


Fig. 2. The higher the interaction strength in a model (IAS increases from left to right), the less representative the Partial Dependence Plot (light thick line) becomes for individual instances represented by their Individual Conditional Expectation curves (dark thin lines).

The less complex the main effects, the better summarizable. In linear models, a feature effect can be expressed by a single number, the regression coefficient. If effects are non-linear the method of choice is visualization [1, 14]. Summarizing the effects with a single number (e.g., using average marginal effects [23]) can be misleading, e.g., the average effect might be zero for U-shaped feature effects. As a by-product of MEC, there is a third option: Instead of reporting a single number, the coefficients of the segmented linear model can be reported. Minimizing MEC means preferring models with main effects that can be described with fewer coefficients, offering a more compact model description.

6 Application: Multi-objective Optimization

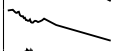
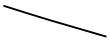
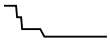
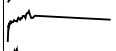
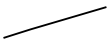
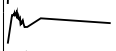
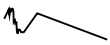
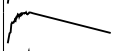
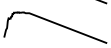
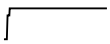
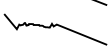
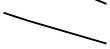
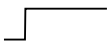
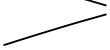
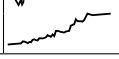
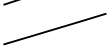
We demonstrate model selection for performance and complexity in a multi-objective optimization approach. For this example, we predict wine quality (scale from 0 to 10) [7] from the wines physical-chemical properties such as alcohol and residual sugar of 4870 white wines. It is difficult to know the desired compromise between model complexity and performance before modeling the data. A solution is multi-objective optimization [12]. We suggest searching over a wide spectrum of model classes and hyperparameter settings, which allows to select a suitable compromise between model complexity and performance.

We used the mlrMBO model-based optimization framework [19] with ParEGO [21] (500 iterations) to find the best models based on four objectives: number of

features used (NF), main effect complexity (MEC), interaction strength (IAS) and cross-validated mean absolute error (MAE) (5-fold cross-validated). We optimized over the space of following model classes (and hyperparameters): **CART** (maximum tree-depth and complexity parameter cp), **s**upport **v**ector **m**achine (cost C and inverse kernel width sigma), **elastic net** regression (regularization alpha and penalization lambda), **g**radient **b**oosted **t**rees (maximum depth, number of iterations), gradient **boost**ed **g**eneralized **a**dditive **m**odel (number of iterations nrounds) and **r**andom **f**orest (number of split features mtry).

Results. The multi-objective optimization resulted in 27 models. The measures had the following ranges: MAE 0.41 – 0.63, number of features 1 – 11, mean effect complexity 1 – 9 and interaction strength 0 – 0.71. For a more informative visualization, we propose to visualize the main effects together with the measures in Table 2. The selected models show different trade-offs between the measures.

Table 2. A selection of four models from the Pareto optimal set, along with their ALE main effect curves. From left to right, the columns show models with 1) lowest MAE, 2) lowest MAE when $MEC = 1$, 3) lowest MAE when $IAS =\leq 0.2$, and 4) lowest MAE with $NF \leq 7$.

	gbt (maxdepth:8, nrounds:269)	svm (C:23.6979, sigma:0.0003)	gbt (maxdepth:3, nrounds:98)	CART (maxdepth:14, cp:0.0074)
MAE	0.41	0.58	0.52	0.59
MEC	4.2	1	4.5	2
IAS	0.64	0	0.2	0.2
NF	11	11	11	4
fixed.acidity				
volatile.acidity				
citric.acid				
residual.sugar				
chlorides				
free.sulfur.dioxide				
total.sulfur.dioxide				
density				
pH				
sulphates				
alcohol				

7 Discussion

We proposed three measures for machine learning model complexity based on functional decomposition: number of features used, interaction strength and main effect complexity. Due to their model-agnostic nature, the measures allow model selection and comparison across different types of models and they can be used as objectives in automated machine learning frameworks. This also includes "white-box" models: For example, the interaction strength of interaction terms in a linear model or the complexity of smooth effects in generalized additive models can be quantified and compared across models. We argued that minimizing these measures for a machine learning model improves its post-hoc interpretation. We demonstrated that the measures can be optimized directly with multi-objective optimization to make the trade-off between performance and post-hoc interpretability explicit.

Limitations. The proposed decomposition of the prediction function and definition of the complexity measures will not be appropriate in every situation. For example, all higher order effects are combined into a single interaction strength measure that does not distinguish between two-way interactions and higher order interactions. However, the framework of accumulated local effect decomposition allows to estimate higher order effects and to construct different interaction measures. The main effect complexity measure only considers linear segments but not, e.g., seasonal components or other structures. Furthermore, the complexity measures quantify machine learning models from a functional point of view and ignore the structure of the model (e.g., whether it can be represented by a tree). For example, main effect complexity and interaction strength measures can be large for short decision trees (e.g. in Table 1).

Implementation. The code for this paper is available at https://github.com/compstat-lmu/paper_2019_iml_measures. For the examples and experiments we relied on the mlr package [5] in R [29].

Acknowledgements. This work is funded by the Bavarian State Ministry of Science and the Arts in the framework of the Centre Digitisation.Bavaria (ZD.B) and supported by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibilities for its content.

References

[1] Apley, D.: Visualizing the effects of predictor variables in black box supervised learning models. arXiv preprint arXiv:1612.08468 (2016)
[2] Apley, D.: Aleplot: Accumulated local effects (ale) plots and partial dependence(pd) plots. CRAN (2017)
[3] Askira-Gelman, I.: Knowledge discovery: comprehensibility of the results. In: Proceedings of the thirty-first Hawaii international conference on system sciences. vol. 5, pp. 247–255. IEEE (1998)
[4] Bibal, A., Frénay, B.: Interpretability of machine learning models and representations: an introduction. In: Proceedings on ESANN. pp. 77–82 (2016)

[5] Bischl, B., Lang, M., Kotthoff, L., Schiffner, J., Richter, J., Studerus, E., Casalicchio, G., Jones, Z.M.: mlr: Machine learning in R. Journal of Machine Learning Research **17**(170), 1–5 (2016)
[6] Casalicchio, G., Molnar, C., Bischl, B.: Visualizing the feature importance for black box models. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 655–670. Springer (2018)
[7] Cortez, P., Cerdeira, A., Almeida, F., Matos, T., Reis, J.: Modeling wine preferences by data mining from physicochemical properties. Decision Support Systems **47**(4), 547–553 (2009)
[8] Dhurandhar, A., Iyengar, V., Luss, R., Shanmugam, K.: TIP: typifying the interpretability of procedures. arXiv preprint arXiv:1706.02952 (2017)
[9] Dua, D., Graff, C.: UCI machine learning repository (2017), http://archive.ics.uci.edu/ml
[10] Fanaee-T, H., Gama, J.: Event labeling combining ensemble detectors and background knowledge. Progress in Artificial Intelligence pp. 1–15 (2013)
[11] Fisher, A., Rudin, C., Dominici, F.: All Models are Wrong but many are Useful: Variable Importance for Black-Box, Proprietary, or Misspecified Prediction Models, using Model Class Reliance. arXiv preprint arXiv:1801.01489 (2018)
[12] Freitas, A.A.: Comprehensible classification models: a position paper. ACM SIGKDD explorations newsletter **15**(1), 1–10 (2014)
[13] Friedler, S.A., Roy, C.D., Scheidegger, C., Slack, D.: Assessing the local interpretability of machine learning models. arXiv preprint arXiv:1902.03501 (2019)
[14] Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Annals of statistics pp. 1189–1232 (2001)
[15] Friedman, J.H., et al.: Multivariate adaptive regression splines. The annals of statistics **19**(1), 1–67 (1991)
[16] Fürnkranz, J., Gamberger, D., Lavrač, N.: Foundations of rule learning. Springer Science & Business Media (2012)
[17] 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), 93 (2018)
[18] Hamidieh, K.: A data-driven statistical model for predicting the critical temperature of a superconductor. Computational Materials Science **154**, 346–354 (2018)
[19] Horn, D., Bischl, B.: Multi-objective parameter configuration of machine learning algorithms using model-based optimization. In: 2016 IEEE Symposium Series on Computational Intelligence (SSCI). pp. 1–8. Ieee (2016)
[20] Huysmans, J., Dejaeger, K., Mues, C., Vanthienen, J., Baesens, B.: An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. Decision Support Systems **51**(1), 141–154 (2011)
[21] Knowles, J.: Parego: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. IEEE Transactions on Evolutionary Computation **10**(1), 50–66 (2006)

[22] Lakkaraju, H., Kamar, E., Caruana, R., Leskovec, J.: Interpretable & explorable approximations of black box models. arXiv preprint arXiv:1707.01154 (2017)
[23] Leeper, T.J.: Interpreting regression results using average marginal effects with R's margins. CRAN (2017)
[24] Molnar, C.: Interpretable Machine Learning (2019), https://christophm.github.io/interpretable-ml-book/
[25] Molnar, C., Bischl, B., Casalicchio, G.: iml: An R package for interpretable machine learning. JOSS **3**(26), 786 (2018)
[26] Philipp, M., Rusch, T., Hornik, K., Strobl, C.: Measuring the stability of results from supervised statistical learning. Journal of Computational and Graphical Statistics **27**(4), 685–700 (2018)
[27] Plumb, G., Al-Shedivat, M., Xing, E., Talwalkar, A.: Regularizing black-box models for improved interpretability. arXiv preprint arXiv:1902.06787 (2019)
[28] Poursabzi-Sangdeh, F., Goldstein, D.G., Hofman, J.M., Vaughan, J.W., Wallach, H.: Manipulating and measuring model interpretability. arXiv preprint arXiv:1802.07810 (2018)
[29] R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2018)
[30] Ribeiro, M.T., Singh, S., Guestrin, C.: Model-agnostic interpretability of machine learning. arXiv preprint arXiv:1606.05386 (2016)
[31] Rüping, S., et al.: Learning interpretable models. Univ. Dortmund (2006), http://d-nb.info/997491736
[32] Schielzeth, H.: Simple means to improve the interpretability of regression coefficients. Methods in Ecology and Evolution **1**(2), 103–113 (2010)
[33] Ustun, B., Rudin, C.: Supersparse linear integer models for optimized medical scoring systems. Machine Learning **102**(3), 349–391 (2016)
[34] Yang, H., Rudin, C., Seltzer, M.: Scalable bayesian rule lists. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 3921–3930. JMLR. org (2017)
[35] Zhou, Q., Liao, F., Mou, C., Wang, P.: Measuring interpretability for different types of machine learning models. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. pp. 295–308 (2018)