KI - Künstliche Intelligenz (2022) 36:125–134
https://doi.org/10.1007/s13218-022-00764-8

TECHNICAL CONTRIBUTION

Explaining Artificial Intelligence with Care

Analyzing the Explainability of Black Box Multiclass Machine Learning Models in Forensics

Gero Szepannek[1] · **Karsten Lübke**[2]

Received: 19 July 2021 / Accepted: 26 April 2022 / Published online: 16 May 2022

Abstract
In the recent past, several popular failures of black box AI systems and regulatory requirements have increased the research interest in explainable and interpretable machine learning. Among the different available approaches of model explanation, partial dependence plots (PDP) represent one of the most famous methods for model-agnostic assessment of a feature's effect on the model response. Although PDPs are commonly used and easy to apply they only provide a simplified view on the model and thus risk to be misleading. Relying on a model interpretation given by a PDP can be of dramatic consequences in an application area such as forensics where decisions may directly affect people's life. For this reason in this paper the degree of model explainability is investigated on a popular real-world data set from the field of forensics: the glass identification database. By means of this example the paper aims to illustrate two important aspects of machine learning model development from the practical point of view in the context of forensics: (1) the importance of a proper process for model selection, hyperparameter tuning and validation as well as (2) the careful used of explainable artificial intelligence. For this purpose, the concept of explainability is extended to multiclass classification problems as e.g. given by the glass data.

Keywords Interpretable machine learning · Multiclass classification · Hyperparameter tuning · Black box algorithms · Partial dependence plots · Explainability · Forensics

1 Introduction

State-of-the art machine learning algorithms in combination with a proper hyperparameter tuning have demonstrated superior performance compared to traditional modelling strategies such as linear or logistic regression or decision trees in many real-world classification and regression problems (cf. e.g. [5, 29]). In turn, the resulting models are often called to be of black box type, i.e. a user does not necessarily understand the behaviour of a model. In the recent past, several popular failures of black box AI systems [21] and regulatory requirements [9, 10, 14] have increased the research interest in explainable and interpretable machine learning. In forensics the impact of decisions based on a failure of a predictive model can be huge as this can directly affect people's life. This puts an even stronger emphasis not only on predictive power on one hand but also on the ability for plausibility checks on the other hand in order to ensure algorithmic fairness [17, 32].

An overview on available methodology of interpretable machine learning linked to specific requirements on the explanation is given in [6]. Moreover, in the paper a general process has been proposed in order to improve transparency, auditability and explainability of machine learning models where the first step of this process consists in a proper model selection and validation. Among the different available approaches of model explanation, partial dependence plots (PDP, [13]) represent one of the most famous methods for model-agnostic assessment of a feature's effect on the model response.

Although PDPs are commonly used and easy to apply they only provide a simplified view on the model and thus risk to be misleading and and in consequence, resulting

✉ Gero Szepannek
gero.szepannek@hochschule-stralsund.de

Karsten Lübke
karsten.luebke@fom.de

[1] Stralsund University of Applied Sciences, Stralsund, Germany

[2] FOM University of Applied Sciences, Dortmund, Germany

Table 1 Glass identification data base: frequencies of the classes

Class	1	2	3	5	6	7
Frequency	70	76	17	13	9	29

interpretations should be undertaken carefully. For this reason in this paper the degree of a model's explainability is investigated on a popular real-world data set from the field of forensics: the glass identification database [8].

The remainder is based on the findings from [6] to the context of the field of forensics: After an introduction of the glass data in Sect. 2 the process of a proper simultaneous hyperparameter tuning and model validation emphasized. Afterwards partial dependence plots are introduced (Sect. 3.1) and applied in order to explain the variable's effects in the model.

In addition, one of main goals of the paper is to highlight the importance of a careful model interpretation. For this purpose the measure of *explainability* Y (Sect. 3.2) is introduced in order to quantify in how far we can trust the explanation given by a PDP. In Sect. 3.3 the concept is extended to multiclass classification problems as in the given example. As it is shown on the glass data in Sect. 4 methods of explainable artificial intelligence provide useful insights but should be used with care. A summary of the results is given in Sect. 5.

2 Glass Identification Database

2.1 Description of the data

The forensic glass identification database is a publicly available data set from the UCI machine learning repository [8] and analyzed by many researchers. An overview of papers that have analyzed the data is given on the corresponding repository website. Most of them solely use the data in benchmark experiments on a broad collection of data sets without any focus on the explicit data set such as [15] (SVMs) and [16] (combined neural network ensemble with knn). In [1] the focus is restricted to the glass data: In addition to the training k nearest neighbour classifiers and a voting approach also some exploratory analysis of the input variables is conducted. Nonetheless, interpretability of the resulting models remains untreated.

The glass data consists of 214 observations collected at the Home Office Forensic Science Laboratory, Birmingham [11] containing examples of the chemical analysis of six different types of glass: 1 (building windows, float processed), 2 (building windows, non float processed), 3 (vehicle windows, float processed), 5 (containers), 6 (tableware) and 7 (headlamps). Vehicle windows, non float processed are not contained were not in the collected data base. The problem is to predict the type of glass on basis of the chemical analysis as given by nine numeric attributes: RI (refractive index), Na (sosium), Mg (magnesium), Al (aluminum), Si (silicon), K (potassium), Ca (calcium), Ba (barium) and Fe (iron). The study of classification of types of glass was motivated by criminological investigation. At the scene of the crime, the glass left can be used as evidence, if it is correctly identified. Table 1 summarizes the frequencies of the six types of glass, i.e. classes. Note that the classes frequencies are different in the sample which should be taken into account in order to avoid biased predictions. The authors are not aware whether the proportions in the data base are representative and restrict to mentioning it here. A comprehensive study on class imbalance correction is given in [5].

2.2 Learning models on the glass data

Proper modelling should to take into account for the following three steps:

1. Model specification selection and performance validation,
2. Parameter tuning and
3. Training of the final model with optimized parameters on the entire data.

For the purpose of this paper the state-of-the art machine learning framework `mlr3` [18] has been used. The steps are described in the following paragraphs.

1. Performance validation benchmark using nested cross validation The first step consists in model selection and validation. For the purpose of this research three algorithms are compared: A single layer feed-forward neural network [25] is tuned w.r.t. the number of neurons in the hidden layer and the weight decay. Note that for reasons of the small number of observations the number of layers has been restricted to

Table 2 Tuning parameters for both algorithms as well as their upper and lower bound for the hyperparameter optimization

Algorithm	Parameter	Lower	Upper	Scale
nn	Decay	0.00001	10	Log
nn	Size	1	20	Linear
rf	Num.trees	64	2048	Log
rf	mtry	1	9	Linear

Optimization on a log scale takes into account for different orders of magnitude in the tuning range

Table 3 Average accuracy of the performance benchmark

	nn	rf	Multinomial	[15]	[16]	[1]
Traininig	0.761	1.000	0.719	Not reported	Not reported	Not reported
Validation	0.650	0.791	0.632	0.663	0.682	0.804

one. This paper does not analyze deep learning this is done by others e.g [7]. As a competitor a random forest is trained as random forests turned out to provide good results in many benchmark studies (cf. e.g. [5, 12, 29]). In addition random forests turned out to be comparatively less sensitive to tuning [23] which is considered to be advantageous given the small number of obervartions in the glass data. Finally, a multinomial log-linear model [25] is computed without tuning which provides a baseline for performance benchmarking.

For tuning of the models the popular hyperband tuning strategy [19] has been used. In contrast to traditonal grid search or random search [2] this strategy is more efficient given a fixed time budget by fastly dropping non-promising parameter combinations [4]. For reasons of the sample size no separate split into training and validation and test data has been undertaken but tenfold cross validation has been used for performance validation and an inner ninefold cross-validation loop for parameter tuning [27]. The tuning parameters and their respective ranges have been chosen based on [24] and are given in Table 2.

Table 3 summarizes the results of the performance benchmark after tuning in terms of the average accuracy on both the (outer) training and validation folds. The results are in line with the conclusion from [6] with regard to the claim of [26] to rely on interpretable models where a proper benchmark is proposed to analyze the benefits of using black box models instead of interpretbable ones: For the glass data a notable gain in predictive power is obtained by using a random forest which supports the results reported in a benchmark study[1] on OpenML [31]. The results further support the hypothesis of random forests being a good baseline choice for benchmarks in many practical applications. All three models show some degree of overfitting the training data which is not unusual given the size of the data and even a natural phenomenon for random forests. The observed accuracy on the validation data in our study is in line with results from other papers that also use tenfold cross validation.[2]

2. Final hyperparameter tuning Note that performance validation in the previous step using tenfold cross-validation combined with nested hyperparameter tuning does not result in a single set of tuned parameters but in a separate set of parameter values for each (outer) loop. For this reason, once provided with a proper estimation of the model's performance a final (unique) hyperparameter tuning is run on the entire data. For this purpose, once again tenfold cross validation is used but this time without nesting. Moreover, in order to allow for a subsequent analysis of the parameters' effects on the performance of the model a 10×10 grid search is used. The resulting heat maps in Figure 1 show the average validation accuracy over the ten validation folds as a function of the underlying hyperparameter setting:

With regard to the tunability of the algorithms [22] the figure underlines that neural networks are more sensitive to tuning compared to random forests. For the forest the number of randomly offered variables at each split is the crucial parameter while the number of trees should not be chosen too small. These results are in concordance with [28]. For neural networks the appropriate choice of the weight decay parameter is of strong importance which should be chosen neither too large nor too small. Despite the small sample size the optimal number of 18 neurons in the hidden layer results in a comparatively high number of weights to be trained. Figure 2 shows rules for chosing the parameters as obtained by a regression tree [30] with the validation accuracy as target variable as a function of the hyperparameters. Table 4 lists the tuned parameters.

3. Retraining the final model on the entire data In order to obtain the best performance final models are trained on the entire data of all 214 observations as a training set with the tuned parameters identified in the previous step.

3 Explainability

3.1 Partial dependence plots

As a central aim of the paper consists in investigating the explainability of machine learning models by partial dependence plots these plots are introduced here. Partial dependence plots (PDP) denote one of the most popular model-agnostic approaches for the purpose of understanding feature effects and go back to [13]. These plots can be easily used to visualize how far one (or several) features impact the outcome of a model. Moreover, as they are model-agnostic PDPs can be computed for arbitrary models, here denoted by $\hat{f}(x)$.

The vector of predictor variables $x = (x_s, x_c)$ is subdivided into two subsets x_s and x_c and the partial dependence function is given by

[1] https://www.openml.org/t/40.

[2] Supplementary R code is available at https://github.com/g-rho/explainability-glass.

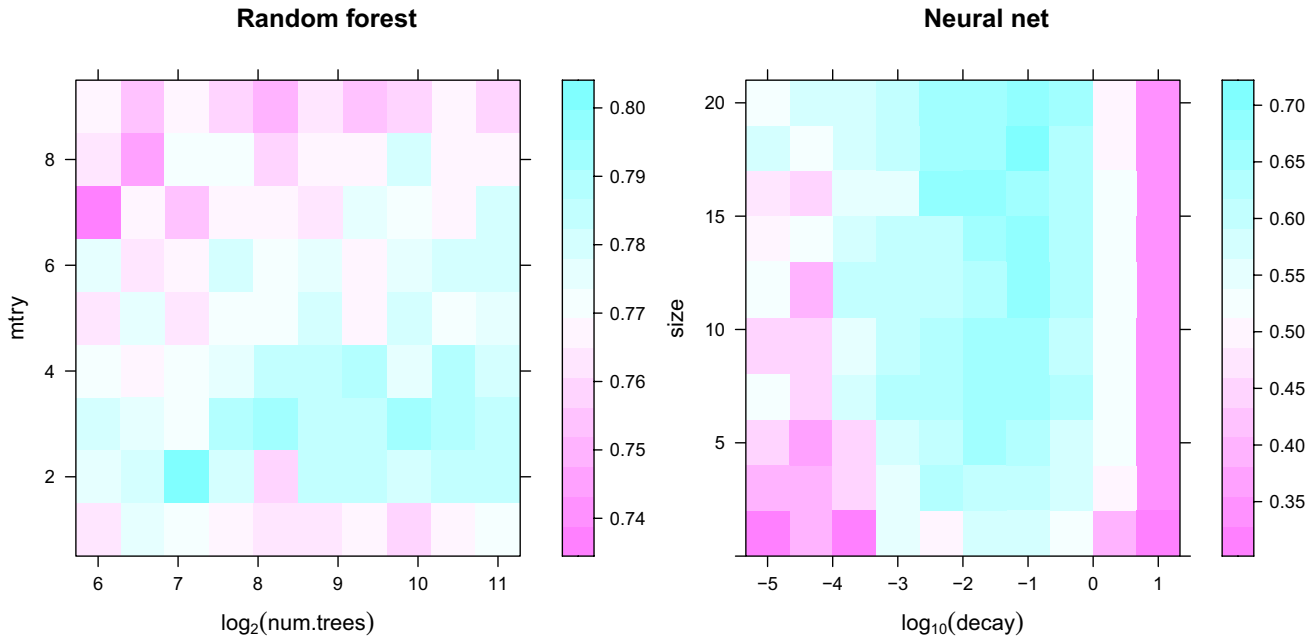

Fig. 1 Effect of the tuning parameters on the validation accuracy. Please note the different colour scales on both heat maps

Fig. 2 Rules for choosing the parameters: Regression trees of the predictive performance as a function of the tuning parameters trained on the data from the hyperparameter tuning

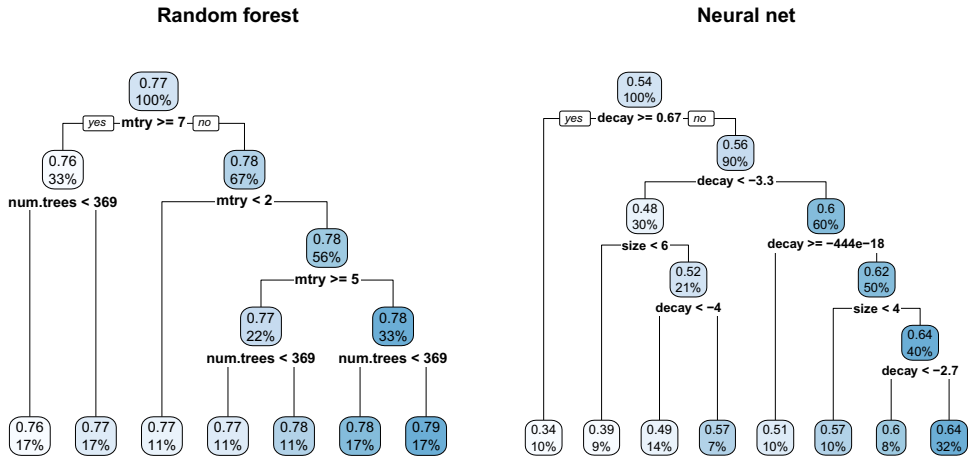

Table 4 Optimal parameters after tuning

Algorithm	Parameter	Values
nn	Decay	0.10
nn	Size	18
rf	Num.trees	138
rf	mtry	2

$$PD_s(X) = PD_s(X_s) = \int \hat{f}(X_s, X_c) dP(X_c), \tag{1}$$

i.e. it computes the average prediction given the variable subset X_s takes the values x_s. In general, partial dependence functions can be computed for variable subsets x_s of any dimension but their visualization however is limited to 1D or 2D. For this reason, partial dependence plots are not able to uncover high order interactions and some relevant information on the model may be missing in the plot.

In practise, for a data set with n observations the partial dependence curve is estimated by

$$\widehat{PD}_s(x) = \widehat{PD}_s(x_s) = \frac{1}{n} \sum_{i=1}^{n} \hat{f}(x_s, x_{ic}). \tag{2}$$

For classification models the prediction $\hat{f}(x)$ usually consists in the predicted posterior probabilities. Figure 3 shows the partial dependence profiles for a multiclass classification problem where each coloured line corresponds to one class.

Fig. 3 PDP for all variables and classes of the random forest model

3.2 A measure of explainability

In order to quantify in how far the representation given by a partial dependence function is able to explain a model we start with a perfect explanation: In this case for all points of the data the PD will be equal to the predictions by the model. A scatterplot of both will have all points on the diagonal (cf. Fig. 4).

The confidence in an explanation given by a partial dependence plot can be measured by the differences between the partial dependence function $PD_s(X_s)$ and the model's predictions. A natural approach to quantify these differences is given by computing the average squared error:

$$ASE(PD_s) = \int \left(\hat{f}(X) - PD_s(X)\right)^2 dP(X_s). \tag{3}$$

Remarkably the *ASE* does not calculate the error between model's predictions and the observations but between the PD and the model's predictions.

Note that for $X_s = X$ the partial dependence function $PD_s(X)$ corresponds to $\hat{f}(X)$ and in the other extreme, for the variable subset $s = \emptyset$, i.e. $X_c = X$, this will end up in

$$PD_\emptyset(X) = PD_\emptyset = \int \hat{f}(X) dP(X), \tag{4}$$

which is independent of X and corresponds to the constant average prediction of the model as estimated by:

$$\widehat{PD}_{\emptyset}(x) = \widehat{PD}_{\emptyset} = \frac{1}{n}\sum_{i=1}^{n}\hat{f}(x_i). \tag{5}$$

In order to allow for an interpretation a PDP's $ASE(PD_s)$ can be benchmarked against the $ASE(PD_{\emptyset})$ of the naive constant average prediction $PD_{\emptyset}$:

$$ASE(PD_{\emptyset}) = \int \left(\hat{f}(X) - PD_{\emptyset}\right)^2 dP(X_s). \tag{6}$$

Finally, a comparison of both quantities, $ASE(PD_s)$ and $ASE(PD_{\emptyset})$ can be used to define the **explainability** Y of any black box model $\hat{f}(X)$ by a partial dependence function $PD_s(X)$ via the ratio

$$Y(PD_s) = 1 - \frac{ASE(PD_s)}{ASE(PD_{\emptyset})}. \tag{7}$$

An Y close to 1 means that a model is well represented by a PDP and the smaller it is the less of the model's predictions are explained in the PDP. Plug-in estimates can be used to calculate the ratio in (7) and quantify the explainability of a PDP for a given model:

$$\hat{Y}(PD_s) = 1 - \frac{\widehat{ASE}(PD_s)}{\widehat{ASE}(PD_{\emptyset})} \tag{8}$$

where $ASE(PD_s)$ (3) is estimated by

$$\widehat{ASE}(PD_s) = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{f}(x_i) - \widehat{PD}_s(x_i)\right)^2 \tag{9}$$

and $ASE(PD_{\emptyset})$ (6) by

$$\widehat{ASE}(PD_{\emptyset}) = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{f}(x_i) - \widehat{PD}_{\emptyset}\right)^2. \tag{10}$$

Identification of the most explainable variables: The measure of explainability Y can be further used to compare different variables with regard to in how far they serve to explain a black box model by their PDP. In addition, a forward variable selection can be used in order to maximally explain a model with as few variables as possible, as shown in Algorithm 1.

Algorithm 1 $\hat{Y}$ based forward variable selection to maximize explainability.

```
Initialize X_s = ∅ and X_c = X.
repeat
    for all variables X_j ∈ X_c do
        X_s^candidate = X_s ∪ X_j
        Compute Ŷ(X_s^candidate)
    end for
    Determine X_j* that maximizes Ŷ(X_s^candidate).
    Set new X_s = X_s ∪ X_j* and X_c = X_c \ X_j*
until X_c ≠ ∅
```

3.3 Multiclass extension of explainability

The measure of explainability as introduced in the previous section has been proposed for scalar numeric predictions $\hat{f}(x)$. In the common case of binary classification these predictions are often given in terms of $\hat{f}(x) := \hat{P}(Y = 1|x)$ the posterior probability of the event of interest $Y = 1$ such as e.g. the occurence of a disease or the churn of a customer. In case of multiclass classification $\hat{f}(x)$ is no longer scalar but a vector consisting of different elements $\hat{f}_k(x) := \hat{P}(Y = k|x)$ for each class k where $\sum_{k=1}^{K}\hat{f}_k(x) = 1$ with K being the number of classes. In this case the explainabilty Y has to be computed for each class separately. In Sect. 4.2 it can be seen that the results for the different variables may differ considerably in terms of the explainability of a variable and the class of interest. For this purpose in the following an extension of explainability for multiclass classification is introduced which is computed simultaneously over all classes.

As a preliminary remark note that the interpretation of a difference of $\left(\hat{f}_k(X) - PD_{k,s}(X_s)\right)$ depends on the posterior probability $\hat{f}_k(x)$, where $PD_{k,s}(X_s)$ denotes the partial dependence function (Eq.1) for class k in a multiclass setting.

The χ^2 statistic $X^2 = \sum_{k=1}^{K}\frac{\left(\hat{f}_k(X) - PD_{k,s}(X_s)\right)^2}{\hat{f}_k(X)}$ compares the squared differences of expected and observed probabilities relative to their magnitude which can be used instead of the original squared differences $\left(\hat{f}(X) - PD_s(X)\right)^2$ for computation of the ASE (Eq. 3) if only one single prediction is considered. We use a $\chi^2_{(K-1)}$ distribution to quantify the observed differences:

$$\hat{Y}_i^{MC} = 1 - F_{\chi^2_{(K-1)}}(X_i^2). \tag{11}$$

This is a measure for the explanation of observation x_i of the data where X_i^2 is the value that the χ^2 statistic takes for observation x_i. $Y_i^{MC} = 1$ represents perfect explanation of the model by the K PDPs and decreasing values correspond to lower explanation. Finally, the multiclass extension of explainability can be obtained by averageing over all individual explanations $\hat{Y}_i^{MC}$:

$$\hat{Y}^{MC} = \frac{1}{n}\sum_{i=1}^{n}\hat{Y}_i^{MC}. \tag{12}$$

Table 5 Explainability $\hat{Y}$ of the PDP for all variables and classes

Class	1	2	3	5	6	7
RI	0.240	0.108	− 0.171	0.037	0.034	0.018
Na	0.050	0.083	0.018	0.006	0.152	0.152
Mg	0.265	0.026	0.104	0.246	0.187	0.152
Al	0.306	0.084	− 0.006	0.103	− 0.010	0.186
Si	0.016	0.025	0.041	0.016	0.015	0.016
K	0.053	0.092	0.008	0.023	0.247	0.029
Ca	0.070	0.205	0.058	0.178	0.025	0.016
Ba	0.044	0.066	0.021	0.002	− 0.024	0.466
Fe	− 0.001	0.022	0.000	0.002	0.007	0.007

4 Aplication to the glass data

4.1 Partial dependence plots

For the remainder of the paper we restrict on the final forest model which showed to be of the best predictive accuracy. Figure 3 shows the partial dependence plots (cf. Sect. 3.1) for all variables and classes of the random forest model using the `DALEX` framework [3]. Note that all predictied posterior probabilities sum up to one which results in the observed scale of the plots.

It can be clearly recognized that the random forest identifies nonlinear dependencies between the input variables and the target: The PDP identifies some prominent nonlinearities, e.g. an increasing concentration of Aluminimum (Al) distinguishes float (blue) vs. non-float processed (cyan) glass from buildings and an increasing content of barium (Ba) strongly increases the probability of a glass to be of class 7 (red, headlamps). A high concentration of natrium is comparatively less prominent in glass from buildings (class 1 and 2) and containers (class 5) but enhance the chance of a glass to be to be tableware (class 6, yellow) or headlamp (class 7, red). High concentration of magnesium only seldomly occurs for both container glass (class 5, purple) and tableware (class 6, yellow). In the following we will analyze how confident we can be in these observed functional dependencies.

4.2 Explainability of the plots

Table 5 shows the explainability $\hat{Y}$ of the partial dependence plots for all variables and all classes. It can be seen that e.g. for glass type class 1 the variable Aluminum has the most explainable PDP. Nonetheless it explains only 30.6% of the variation within the model's predictions. Figure 4 illustrates this: For a perfect match of the partial dependence function and the model predictions all points should be on the diagonal or at least close to it which is clearly not the case here. Note that the abscissa covers a smaller range of values which is obvious as the partial dependence represents an average. Further note the differences between the classes in Table 5: Different classes are differently well explained by different variables.

An analysis of the PDPs in the previous subsection turned out that e.g. an increasing concentration of aluminimum (Al) distinguishes float vs. non-float processed glass from buildings (class 1 and 2). An additional consideration of the corresponding values of $\hat{Y}$ turns out that the PDP of aluminium is much more explainable for class 1 than for class 2. On the other hand the observed interpretation of the effect of Magnesium on the prediction of container glass and tableware (class 5 and 6) is supported by their comparatively high values of $\hat{Y}$. So far, no thresholds or rules of thumb rules for explainability are available in the literature.

Table 6 contains a summary of the most explainable two-dimensional PDP after variable selection as described in the

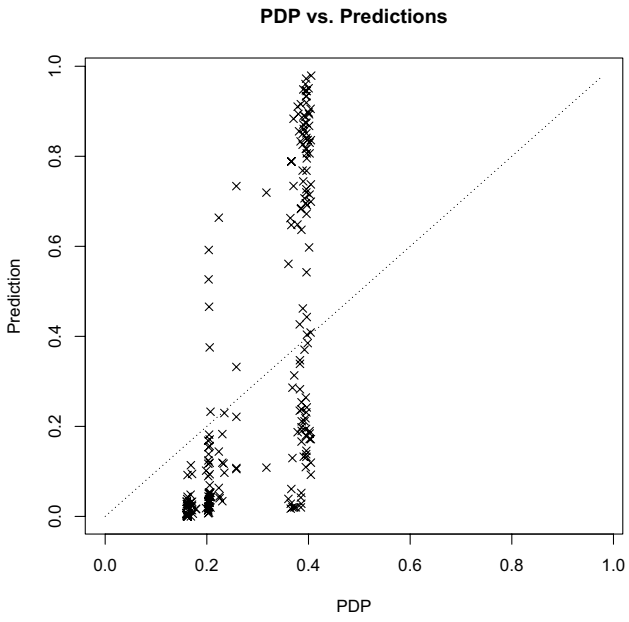

Fig. 4 Difference between PDP of aluminum and the model's predictions for class 1

Table 6 Variable selection of the two most explainable variables for all classes

Class	1	2	3	5	6	7
Step 1	Al	Ca	Mg	Mg	K	Ba
Step 2	Mg	RI	Ca	Na	Ba	Al
$\hat{Y}$	0.523	0.350	0.226	0.470	0.324	0.643

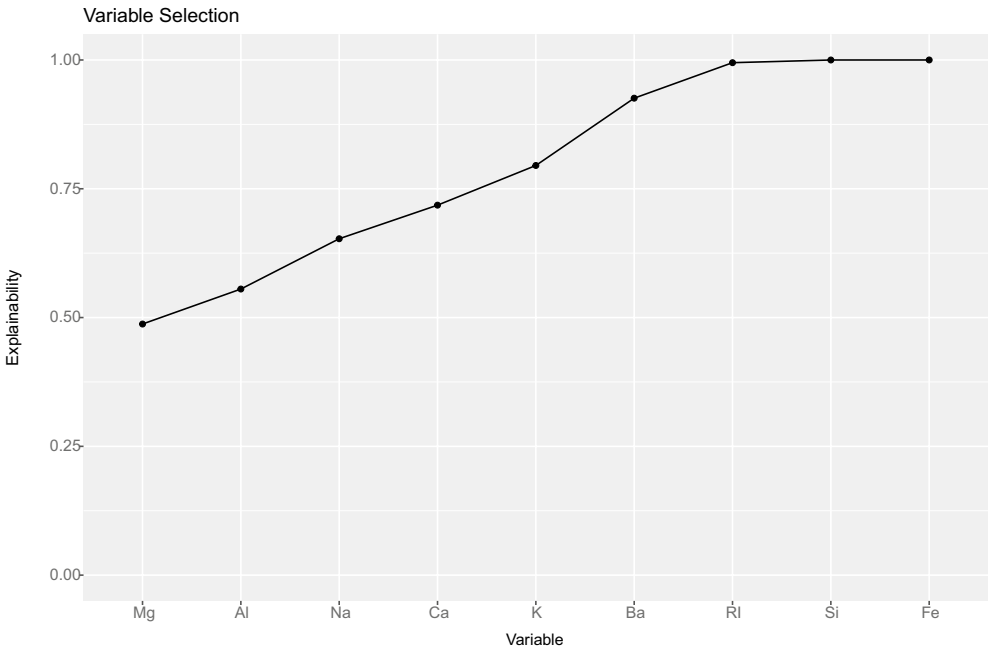

Fig. 5 Variable selection silmutaneously taking into account for all classes

previous Section together with the resulting explainability. Some of the classes (1, 5, 7) are better explained $\hat{Y} \sim 0.5$ or greater while the other classes aren't. Further note that the selected variables differ among the classes. So far there is no graphical representation for partial dependence functions of $dim(X_s) > 2$.

In addition, $\hat{Y}^{MC}$ based variable selection is used to summarize the explainabilities of the PDPs for all classes. Figure 5 shows the results of the variable selection according to algorithm 1. With the two most explainable variables Mg (magnesium) and Al (aluminium) a multiclass explainability of 0.5553 is obtained: The random forest model can be explained up to some extent but the explanation is not perfect so some information on the model is missing in the plots. For the two most explainable variables two-dimensional PDPs are computed for all classes using the `iml` framework [20]. The corresponding heat maps are given in Fig. 6. From the color scale it can be easily recognized that the classes have different prior probabilites. Such two-dimensional PDPs allow to detect interactions but unfortunately no visualization for PDPs of $dim(X_s) > 2$ is possible and thus higher order interactions as they might have been identified by the underlying model stay hidden. Both measures $\hat{Y}$ and $\hat{Y}^{MC}$ help to quantify the degree of explanation of the model by its corresponding partial dependence plots.

5 Summary

The impact of decisions based on the use of predictive models in forensics underlines the importance of both high accuracy as well as an available interpretation of the model. For this purpose, the use of state-of the art machine learning in the field of forensics has to circumvent two competing challenges: a proper model selection and validation as well as the explanation of the model for the sake of plausibility checks which are hard to achieve as the models are often of black box nature. By means of the example of the public glass identification data base both aspects are discussed: The process of model selection and validation has to cover the following steps:

1. Model specification selection and performance validation,
2. Parameter tuning and

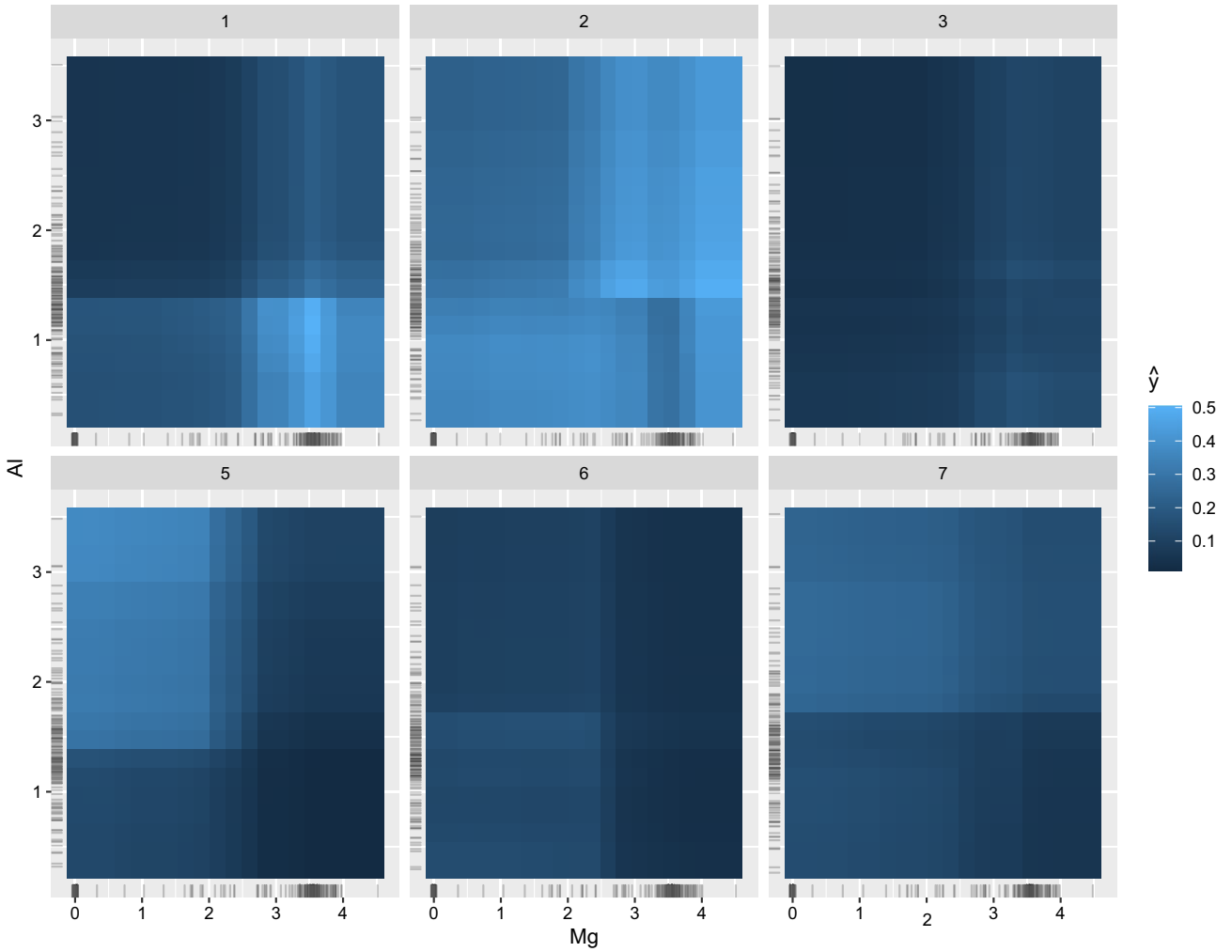

Fig. 6 2D PDPs for all classes and the two most explainable variables Mg (abscissa) and Al (ordinate). Each heat map visualizes the average predicted posterior probabilities for the corresponding class as given by the title

3. Training of the final model with optimized parameters on the entire data.

For the example within this paper random forests as well as neural networks have been tuned with respect to their hyperparameters. The results confirm those from [28] that random forests show good performance which is comparatively insensitive to tuning as compared to other algorithms. Compared to an interpretable linear baseline model strong improvements in predictive power are achieved. These results do confirm the proposed approach in [6] of carefully investigating the benefits of tuned black box models vs. interpretable challengers. The small size of the glass data base has to be emphasized. Of course, with an increased amount of available data the best performing model class might be another one.

For the purpose of model interpretation partial dependence plots provide a well-known and popular tool for model interpretation. As it has been shown in this paper it is important to be aware that the resulting plots only partially explain the model. The measure of explainability can be used to quantify how much of the model's predictions are visualized by a partial dependence plot. The specific situation of multiclass classification as it is given by the glass identification data base raises the need for a multiclass extension of the concept of explainability. A measure that can be used for this purpose has been proposed based on the χ^2 statistic and applied to the random forest model on the glass data. So far there are no thresholds or thumb rules for the explainability to support the confidence in a model explanation. In application contexts as delicate as the field of forensics one should restrict to careful interpretations and decisions.

Funding Open Access funding enabled and organized by Projekt DEAL. Funding was provided by Stralsund University of Applied Sciences.

References

1. Aldayel MS (2012) K-nearest neighbor classification for glass identification problem. In: 2012 internatioal conference on computer systems and industrial informatics, pp 1–5. https://doi.org/10.1109/ICCSII.2012.6454522
2. Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. JMLR 13:281–305
3. Biecek P (2018) DALEX: explainers for complex predictive models in R. J Mach Learn Res 19(84):1–5
4. Bischl B, Binder M, Lang M, Pielok T, Richter J, Coors S, Thomas J, Ullmann T, Becker M, Boulesteix AL, Deng D, Lindauer M (2021) Hyperparameter optimization: foundations, algorithms, best practices and open challenges. arXiv:2107.05847. Accessed 19 July 2021
5. Bischl B, Kühn T, Szepannek G (2016) On class imbalance correction for classification algorithms in credit scoring. In: Lübbecke M (ed) Operations research Proceedings 2014. Springer International Publishing, pp 37–43
6. Bücker M, Szepannek G, Gosiewska A, Biecek P (2021) Transparency, auditability and explainability of machine learning models in credit scoring. J Oper Res Soc. https://doi.org/10.1080/01605682.2021.1922098
7. Dombrowski AK, Anders CJ, Müller KR, Kessel P (2022) Towards robust explanations for deep neural networks. Pattern Recognit. https://doi.org/10.1016/j.patcog.2021.108194
8. Dua D, Graff C (2017) UCI machine learning repository. http://archive.ics.uci.edu/ml. Accessed 19 July 2021
9. EU Expert Group on AI (2019) Ethics guidelines for trustworthy AI. https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai. Accessed 19 July 2021
10. European Commission (2020) On artificial intelligence—a European approach to excellence and trust. https://ec.europa.eu/info/publications/white-paper-artificial-intelligenceeuropean-approach-excellence-and-trust_en. Accessed 19 July 2021
11. Evett IW, Spiehler EJ (1987) Rule induction in forensic science. In: KBS in Government, pp 107–118
12. Fernández-Delgado M, Cernadas E, Barro S, Amorim D (2014) Do we need hundreds of classifiers to solve real world classification problems? J Mach Learn Res 15(1):3133–3181
13. Friedman JH (1991) Multivariate adaptive regression splines. Ann Stat 19(1):1–67. https://doi.org/10.1214/aos/1176347963
14. Goodman B, Flaxman S (2017) European Union regulations on algorithmic decision-making and a "right to explanation''. AI Mag 38(3):50–57. https://doi.org/10.1609/aimag.v38i3.2741
15. Hsu CW, Lin CJ (2002) A comparison of methods for multiclass support vector machines. IEEE Trans Neural Netw 13(2):415–425. https://doi.org/10.1109/72.991427
16. Jiang Y, Zhou ZH (2004) Editing training data for knn classifiers with neural network ensemble. In: Yin FL, Wang J, Guo C (eds) Advances in neural networks—ISNN 2004. Springer, Berlin Heidelberg, pp 356–361
17. Kusner M, Loftus J (2020) The long road to fairer algorithms. Nature 534:34–36
18. Lang M, Binder M, Richter J, Schratz P, Pfisterer F, Coors S, Au Q, Casalicchio G, Kotthoff L, Bischl B (2019) mlr3: a modern object-oriented machine learning framework in R. J Open Source Softw. https://doi.org/10.21105/joss.01903
19. Li L, Jamieson K, DeSalvo G, Rostamizadeh A, Talwalkar A (2017) Hyperband: a novel bandit-based approach to hyperparameter optimization. J Mach Learn Res 18(1):6765–6816
20. Molnar C, Bischl B, Casalicchio G (2018) iml: an r package for interpretable machine learning. JOSS 3(26):786. https://doi.org/10.21105/joss.00786
21. O'Neil C (2016) Weapons of math destruction: how big data increases inequality and threatens democracy. Crown Publishing Group, New York,
22. Probst P, Boulesteix AL, Bischl B (2019) Tunability: importance of hyperparameters of machine learning algorithms. J Mach Learn Res 20(53):1–32
23. Probst P, Wright MN, Boulesteix AL (2019) Hyperparameters and tuning strategies for random forest. WIREs Data Min Knowl Discov 9(3):e1301. https://doi.org/10.1002/widm.1301
24. Richter J (2020) mlrhyperopt parconfigs. http://mlrhyperopt.jakob-r.de/parconfigs. Accessed 19 July 2021
25. Ripley B (2021) nnet: Feed-forward neural networks and multinomial log-linear models. R package version 7.3-116. https://CRAN.R-project.org/package=nnet. Accessed 19 July 2021
26. Rudin C (2019) Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. https://arxiv.org/abs/1811.10154. Accessed 19 July 2021
27. Simon R (2007) Resampling strategies for model assessment and selection. Springer, New York, pp 173–186
28. Szepannek G (2017) On the practical relevance of modern machine learning algorithms for credit scoring applications. WIAS Rep Ser 29:88–96. https://doi.org/10.20347/wias.report.29
29. Szepannek G, Gruhne M, Bischl B, Krey S, Harczos T, Klefenz F, Dittmar C, Weihs C (2010) Perceptually based phoneme recognition in popular music. In: Locarek-Junge H, Weihs C (eds) Classification as a tool for research. Springer, Berlin, pp 751–758
30. Therneau T, Atkinson E (1997) An introduction to recursive partitioning using the rpart routines. Technical report 61, Division of Biostatistics, Mayo Foundation
31. Vanschoren J, van Rijn JN, Bischl B, Torgo L (2013) Openml: networked science in machine learning. SIGKDD Explor 15(2):49–60. https://doi.org/10.1145/2641190.2641198
32. Verma S, Rubin J (2018) Fairness definitions explained. In: Proceedings of the international workshop on software fairness, FairWare '18. ACM, New York, NY, USA, pp 1–7. https://doi.org/10.1145/3194770.3194776