Visual Tools for Understanding Regression Black-Box Models

Inês Areosa Rodrigues
maria.rodrigues@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

November 2019

Abstract

Lack of transparency has become a great barrier to the widespread adoption of machine learning in many areas of human society, despite the outstanding performance of recent algorithms in terms of accuracy. When accounting for important and costly decisions, end users need to understand the model to be able to rely on the predictions. In that regard, explaining black-box models has become a hot topic in Machine Learning. This paper develops a novel approach to inspect the estimated risks of using a black-box regression model for a given test case. We describe, evaluate and propose tools that visually convey the relationship between the expected error and the values of a predictor variable. Moreover, we illustrate the usefulness of our tools by applying them and other state-of-the-art methods to a concrete real world case study with high socioeconomic impact: understanding factors that drive the fishing effort around Large Scale Marine Protected Areas.

Keywords: explainability, black box model, regression, performance , Large Scale Marine Protected Areas

1. Introduction

Sophisticated machine learning algorithms developed recently have reached a complexity level that inherently hinders their functioning. As these models begin driving important and costly decisions, end users have been pressuring for explainability and transparency. In fact, the lack of transparency is currently one of the largest obstacles to the wide-scale adoption of machine learning. In this context, implementing explainable models and understanding black box models has become one of the hot topics in Artificial Intelligence (AI) research.

There are plenty of methods one can use for better understanding the behaviour of a model. In this work, we address the explanation of (black-box) regression models through the usage of visual methods, since these are more adequate for conveying information to end users with reduced technical background. Most existing explainability work analyses the output (predicted values) of the algorithm. However, we claim that explaining the performance (prediction error) of the model is also of high relevance, particularly when the predictions drive costly decisions. We will then focus on this particular aspect, proposing three tools that provide insight into the relationship between the prediction error of the models and the values of the predictor variables. This approach helps the end user assessing the risks of using the regression model in certain domains, as well as explaining the reasons behind some performance degradation.

Finally, to showcase the competence of explainability tools, we analyse the fishing effort of vessels around different areas of the globe near to Large Scale Marine Protected Areas (LSMPA) using various methods. Using the proposed tools, we begin by comparing predictive algorithms to select the most suitable for the problem, following with an overview analysis of the performance of the chosen algorithm. Lastly, we employ some selected state-of-the-art interpretability tools to fully understand the impact of a set of environmental, physical and economic factors on the fishing effort.

This paper is organised as follows. In Section 2 we provide an overview of the existent tools. Our proposals are described in Section 3 and the results of the real world application are discussed in Section 4. Finally, the main conclusions are presented in Section 5.

2. Background

The main innovations and endeavours in explainable AI can be represented by the considerable number of interpretability tools that have been being suggested recently [1, 12], which intend to understand the influence of the inputs in the predicted outcome, answering to the problem of interpretability. These methods can be distinguished between global and local explanations, depending on whether the functioning of a model is described in broad terms or if the explanation concerns a prediction for a specific instance. Moreover, existing tools approach the problem from different perspectives. For instance, feature importance methods try to attribute a score for the influence of each feature on the prediction function [7, 9]. Other tools try to visually show the relationship between the values of a predictor variable and the output of a model and can even be extended to display the interaction between two predictors [4, 10].

Most existing work focuses on analysing the predictions of the models. However, we claim that in order to trust a prediction it is also crucial to provide an assessment of the risk of the model, and on this necessity lies the problem of accountability. The evaluation of a regression model hinges on the differences between the true and predicted values and can be executed using scalar or graphical metrics. The former methods, most commonly used, quantify an estimate of the expected error, using approaches such as the Mean Squared Error and the Median Error [13, 21]. Other approaches compare the quality of several models [2, 20] by estimating the loss of information. However, these methods provide a single metric for the entire model, concealing information if certain predicted values tend to be more error prone. Graphic metrics provide a different perspective on the analysis of the model, informing about the changes in the performance for different operating conditions, as are example the REC curves [5] and surfaces [22]. REC curves plot the error tolerance versus the percentage of points predicted under that same tolerance, representing an estimation of the cumulative distribution function of the error of the model. REC surfaces, in turn, add the target values to this graphic.

The existing accountability tools only address the overall error or the error tolerance in respect to the target value. These methods assess the model as a whole, not considering that different conditions might lead to distinct performance behaviours as well as not establishing a relationship with the predictor's values. Thus, this will be the main distinguishing factor of our proposals.

3. Explaining the Performance of the Black-Box

In light of previous research, we concluded that there has been a great development in the field of interpretability methods to the detriment of investigation of accountability methods, which have yet not seen any recent innovation considering regression tasks.

In this section we describe a set of visual tools that help explaining the performance of different black box regression models. Our approach relates the expected error of the model to the predictor variables values to provide a more detailed analysis of the risk associated with trusting a model for a concrete test case.

Table 1: Data sets used for the experiments.

Data Set	*Nr. Cases*	*Nr. Predictors*
a1	198	11
a2	198	11
a3	198	11
a4	198	11
a6	198	11
a7	198	11
Abalone	4177	8
acceleration	1732	14
availPwr	1802	15
bank8FM	4499	8
cpuSm	8192	12
fuelCons	1764	37
boston	506	13
maxTorque	1802	32
servo	167	4
airfoild	1503	5
concreteStrength	1030	8
machineCpu	209	6

Table 2: Regression algorithms, parameters, and respective used R packages.

Learner	Parameter Variants	R package
Neural Networks (NN)	$size = 10$, $decay = 0.1$, $maxit = 1000$	**nnet** [24]
Support Vector Machines (SVM)	$cost = 10$, $gamma = 0.01$	**e1071** [8]
Random Forests (RF)	$ntree = 1000$	**randomForest** [16]
Gradient Boosting Machines (GBM)	*distribution* = "*gaussian*", $n.trees = 5000$, $interaction.depth = 3$	**gbm** [11]

Throughout this section we will illustrate the proposed tools using experiments carried out on 18 regression data sets, with properties described in Table 1. To guarantee that the tools are model-agnostic and to avoid the existence of model-dependent bias, each data set was modelled using the four distinct predictive learning algorithms described in Table 2. As stated before, our tools hinge on the analysis of the error of a regression model. To ensure the trustworthiness of our results, the prediction error of each data set and model was estimated using 10-fold Cross Validation (CV) with the R package *performanceEstimation* [23]. This procedure allowed us to obtain, for each case in the data set, a reliable estimate of the prediction error of a black-box model we want to study.

Due to the extensive number of data sets, models and predictors, we cannot showcase all the results here. Hence, we will only illustrate some pertinent examples, while the full graphs can be consulted at `http://github.com/inesareosa/MScThesis`. The web page also contains the source code for each tool and to obtain all figures, all implemented in R [18], ensuring full reproducibility of the results and analysis.

3.1. Error Dependence Plot

The first tool to be introduced, the Error Dependence Plot (EDP), shows the relation between the expected error of a single regression model and the values or categories of a predictor.

Obtaining the distribution of the error for each value of a numeric predictor is strenuous, specially for continuous variables and small data sets since each value will probably not repeat many times in the data set. To ensure a reliable visualization, we propose discretizing the numerical predictor variables into meaningful bins. This process allows for the collection of several error values in each bin, therefore enabling the estimation of the error distribution. Ideally, the range of the bins should be selected by an expert in accordance with the analysis goals. Nevertheless, as this knowledge is not always available, we here suggest selecting the bins with the following quantiles of the predictor variable distribution: [0 - 10%] (extremely low values); [10% - 35%] (low values); [35% - 65%] (central values) [65% - 90%] (high values); and [90% - 100%] (extremely high values). Note that this binning process is not required for nominal predictor variables, that have their bins defined by their categories.

Our proposed EDPs show the distribution of the estimated error, in the Y-axis, for each bin of the predictor values, in the X-axis, by grouping all the training cases that match each bin and then showing the respective error distribution through the usage of boxplots. EDPs also display the overall error distribution and a line indicating the median error for comparison, as well as the information of the number of training cases belonging to each bin and the respective percentage of the data set.

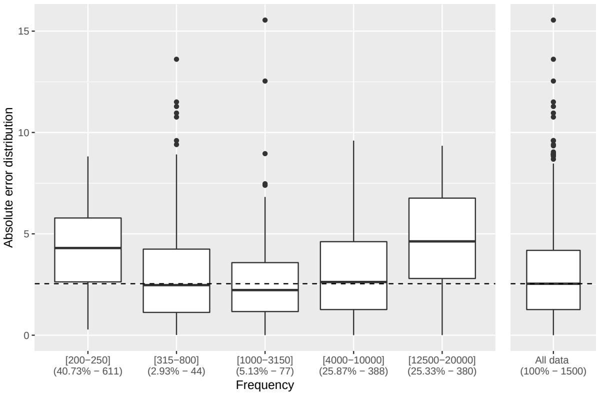


Figure 1: EDP for feature *Frequency* of data set *airfoild* trained with RF.

Figure 1 shows the EDP for the numerical feature *Frequency* of data set *airfoild* when trained with a Random Forest (RF). The feature values were segmented into 5 bins, as shown in the X-axis of the plot, with each boxplot representing the estimated error distribution of the RF in the respective range of values. This illustrative EDP shows that extreme values ($Frequency = [200, 250]$ and $Frequency = [12500, 20000]$) have a considerably distinct estimated error distribution comparing to the overall data, displayed in the rightmost part of the plot. For these values, the Random Forest is expected to have a considerably worse performance. Nevertheless, the EDP also shows that the higher prediction errors of the Random Forest, although rare, occurred in other ranges of the variable *Frequency* where the performance of the model is expected to be much better. This may serve as an alert of the possible influence of other factors in those ranges of *Frequency*.

3.1.1 Bivariate EDPs

EDPs are univariate and thus ignore interactions among the predictors, which may impact the performance of the models. To capture some of these potential interacions we suggest the usage of bivariate EDPs. These graphs are conceptually the same as the EDPs but they show the estimated error distribution for a combination of two predictors. These are obtained with a similar procedure as EDPs, the only difference being that the partition of the errors is made across all possible combinations of bins between both predictors, instead of the bins of a single predictor. For a deeper insight, EDPs can also be adjusted to capture trivariate interactions.

Figure 2 shows an example of a Bivariate EDP for the data set *a7* when trained with a Support Vector Machine (SVM). This case explores the impact of the predictors *season* and $PO4$ on the estimated error of the model. The top left panel shows the overall error distribution of the model without any conditioning of the two predictors. Each of the remaining panels then represent a different bin of $PO4$. For each bin of $PO4$, having the bins of *season* as the X-axis, we show the boxplots of the estimated error distribution for the respective combination of predictor values. This small example allows us to observe that the SVM has a considerably different behaviour for $season = autumn$ when $PO4$ is in the range $[169 - 285.71]$ (high values of $PO4$). We can also observe that for the lowest values of $PO4$ the performance is generally much better, independently of the season. Note that if a combination of values of the two variables does not occur in the training data, the respective boxplot is not shown, as it is example the joint occurrence of extremely high values of $PO4$ and *size=autumn*.

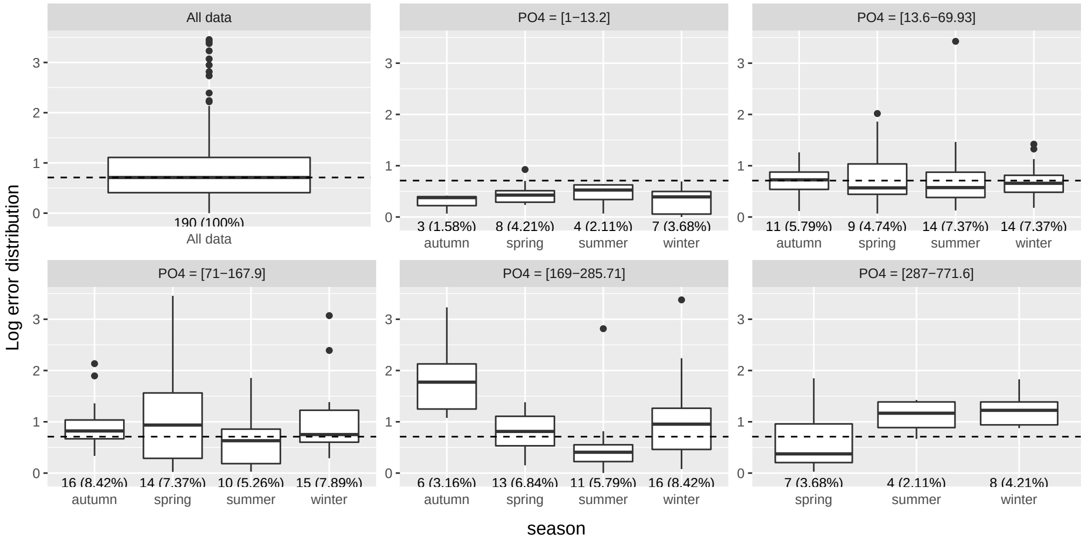


Figure 2: Bivariate EDP for data set *a7* trained with SVM for features *season* and *PO4*.

3.1.2 Evaluating EDPs

To understand how effective EDPs are in anticipating the error of the model in future test cases, we have carried out a visual and a metric evaluation.

For both experiments, each data set in Table 1 was randomly partitioned into a training (70%) and a test set (30%). Using the training set, the models in Table 2 were trained and the respective estimated errors for each instance were computed using CV, with the estimates being used to obtain EDPs. Afterwards, with the models learned in the 70% training set, we obtained the predictions and respective errors on the separate 30% test set left out of the EDP creation. We aim to verify whether the distribution of the errors obtained on the test set is similar to the distribution shown on the EDPs obtained using the CV estimates of the training data.

To facilitate the comparison of the distribution of the errors observed on the test set to the distribution shown on the EDPs, we propose a variant of these graphs where two boxplots are show for each value of the predictor: (i) the original boxplot of the EDP; and (ii) the boxplot of the errors of the model on the test set. Through visual inspection, the end user can verify if the actual error had a similar distribution to the one indicated by the EDP.

The analysis of the experiments carried out with all the data sets in Table 1 lead to the conclusion that EDPs tend to have higher reliability for larger data sets and, in most cases, for bins with considerable representation. This occurs because CV estimates are more effective when the available data samples are sufficiently large. Figure 3 shows the EDP evaluation of a Gradient Boosting Machine (GBM) trained on data set *fuelCons* for feature *Attribute9*. The major disparity between the error distributions is observed for *Attribute9* = [7600-55000], where the number of cases with this value in the training set was of only 8 (0.65% of the training set), and in the test set only 3 cases had this value of the feature. Another bin, with the range *Attribute4*=[3000-3700], presents a smaller discrepancy, having 28 (2.27%) cases on the training set. The other bins, with larger representation, present the best results, with both errors showing a similar distribution, showcasing the reliability of the EDPs information.

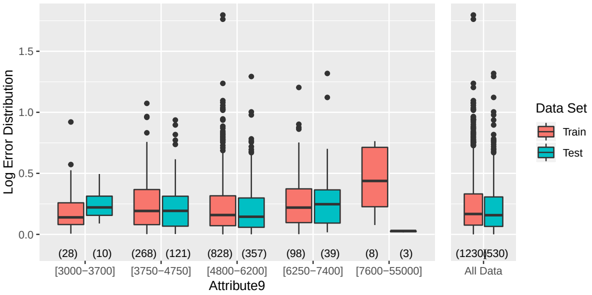


Figure 3: Evaluation of EDP for feature *Attribute9* of data set *fuelCons* trained with GBM.

To further investigate the reliability of the estimates provided by EDPs, we ran a formal test that compares the equality of two continuous distributions: the Anderson-Darling (AD) test [3], which assumes as a null hypothesis that the two samples (each bin of the EDP and the test errors obtained for the same predictor values) are drawn from the same distribution. In our evaluation, rejecting the null hypothesis would then mean that the EDPs did not estimated successfully the error behaviour for that given range of predictor values. The test was conducted using the function *ad.test* from the R package *kSamples* [19], that returns a p-value which should be compared to a significance level α - if p-value$\leq \alpha$, the null hypothesis should be rejected.

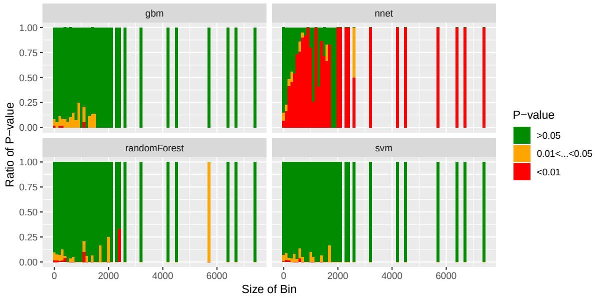


Figure 4: Proportion of p-values against size of bin for each model (GBM, NN, RF and SVM), distinguished if above or below $\alpha = 0.05$ and $\alpha = 0.01$. Bar width of 100 instances.

Figure 4 analyses the resultant AD test p-values against two commonly used significance levels $\alpha = \{0.01, 0.05\}$ for the size of the predictor bins of each data set, for each algorithm. Here we distinguish between the percentage of p-values above $\alpha = 0.05$ (in green), below $\alpha = 0.05$ (in yellow and red) and below $\alpha = 0.01$ (in red), plotting the ratio of p-values for each size of the bin. We observe a particularity for the Neural Networks, in which the CV estimates are clearly the least effective in predicting the error behaviour, as we can conclude by the high percentage of p-values under 0.01. All the other models, specially the GBM and the SVM, show an improvement on the reliability of the estimates with the increase of the size of the bin, strengthening the assumptions made in the previously discussed visual evaluation.

In summary, our experiments show that EDPs obtained using a CV process to estimate the errors of a black box model will generally provide reliable estimates of the expected error of the model for each feature value, if enough data is available for this CV process. However, a special advert should be made to the usage of EDPs with Neural Networks, since their estimates were found not to provide trustworthy results. Note that these results do not arise from a failure of the EDPs but rather from the fact that the estimated CV errors have been observed to be unreliable, due to an higher instability of the NNs that lead to the performance on a training set not being completely replicable on a test set.

3.2. Parallel Error Plots

EDPs have limitations when plotting various variables simultaneously, since these are restricted to display the information for a maximum of three variables at a time. However, most real world problems have many more feature variables and potential interactions between these should not be ignored. From this perspective, we propose an extension of EDPs to a multivariate representation: the Parallel Error Plots (PEPs), which represent the estimated error profile across the range of values of multiple predictors simultaneously through the usage of parallel coordinate plots [14].

Given the limitations of parallel coordinate plots, we decided to split the very high errors from the rest, assuming that the end users are interested in knowing the conditions that lead the models to a unusually bad performance as these might be an indicator of higher risk. Hence, we suggest dividing the errors in two categories: the top 10% errors and the rest. This division is not strict and in fact, the user is free to select another criterion and even advised to so if dealing with extremely large data sets, where the percentage should be adjusted to avoid jeopardizing visualization.

PEPs plot each feature variable in the X-axis, represented by a vertical bar that results from uniforming the scale of each variable. The uniformization of PEPs map the original range of each feature into a [0,1] scale, with 0 corresponding to the minimum and 1 to the maximum values of the variable in the data set. Mapping all feature values to this uniform scale supports the display of all values on the same Y-axis. Using this scale, each instance of a data set is then represented by a line that crosses each vertical bar in the respective scaled value of the predictor. Additionally, PEPs color the line of each case according to the respective estimated error: if the model had a very high expected error (by default on the top 10%) when forecasting some case the corresponding line is shown in red, otherwise the line is drawn in grey. This enables the end user to easily detect some patterns and overall tendencies concerning the conditions in terms of the predictors that lead to higher prediction errors.

We advise ordering the predictors in the X-axis by a score of feature relevance, using feature importance methods. This way end users can easily confirm whether the lowest performance is explained by the most important features. Here, we ordered the predictors by importance using the function *varImp* from the R package *caret* [15], which calculates the variable importance through model-specific methods.

Figure 5 depicts the PEP of a RF trained on data set *a1*, with all predictors of the data set ordered from left to right in increasing order of estimated feature importance. PEPs help in identifying interesting patterns concerning the largest errors of the models. In fact, with this plot it can be observed that the largest errors of the Random Forest

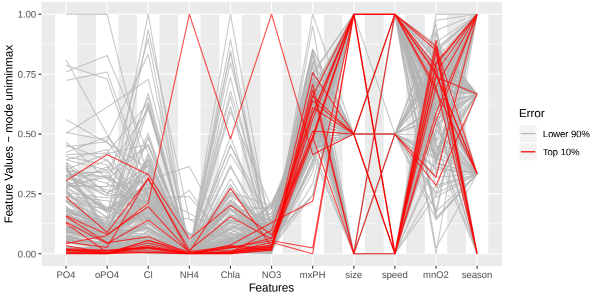


Figure 5: Parallel Error Plot of Random Forest for dataset *a1*.

occur for the cases in which *PO4, oPO4, Cl, NH4, Chla* and *NO3* have lower values of their range and when *mnO2* has higher values . On the other hand, the highest errors do not seem to be strongly correlated with the values of the variables *season*, *size* and *speed*. This type of information can be of great value when deciding whether we can trust the prediction of the Random Forest for a new test case of this problem.

PEPs present some limitations when outliers occur in the predictors range, since this leads to a compression of the other values. This limitation can be addressed by using other methods of making the scales of the variables uniform, that are robust to outliers. Furthermore, with a large data set, the visualisation might get confusing. In these cases, one can either randomly subset the data set to help interpretation or pick a lower percentage for defining the top errors. Finally, the visualization provided by PEPs may also suffer from an excessive number of predictors. In these cases we can opt for showing only a subset of the predictors, which could be determined by scores of feature relevance.

3.3. Multiple model Error Dependence Plots

EDPs and PEPs allow to understand the conditions in terms of predictor values that lead to a different predictive performance of one model. In this section we argue that it is also relevant to do this analysis with the goal of comparing different models on the same problem, as it would allow to make model selection for specific test cases based on their predictor values. With this goal we propose the Multiple model Error Dependence Plots (MEDPs), which develop EDPs to further analyse multiple models simultaneously, across the range of a predictor.

In similarity to the EDPs, we compartmentalize the feature of interest using the process described in Section 3.1. Then, for each bin within the predictor values, the error boxplots of each model representing the estimated error distribution (obtained using CV) are arranged side by side, enabling the end user to compare the performance of the models across the domain of the chosen predictor. In order to facilitate comparisons, the MEDPs show the overall error distribution and a dashed line representing the median predicted error for each model. Through visual inspection of the estimated error behaviour, this tool helps finding the model most suitable for any particular test case given its feature values.

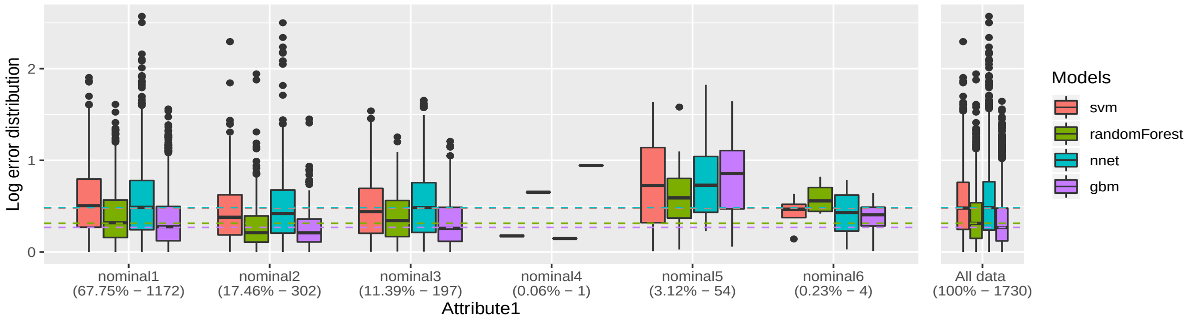
As EDPs, MEDPs present a bivariate variant that allows for the identification of interactions between predictors, using the same visualisation method but showing several boxplots (one for each model) for each combination of bins.

MEDPs can be a useful tool to decide between two models with a similar overall performance or to identify whether the model with the best global performance is outperformed for a certain range or category of predictor variables. As an illustrative example, take the MEDP on the *acceleration* data set for *Attribute1* (Figure 6), in which the best overall performance is achieved with the GBM, as seen in the right part of the plot. Analysing the performance of all 4 models for each category of *Attribute1*, one can conclude by the expected error distribution that this model in fact underperforms when the feature *Attribute1=nominal5*. This plot then indicates that, if operating in domains of *nominal5*, the GBM is not as reliable as the other models, in contrary to the what the overall performance illustrates. Note that the bin *Attribute1* = *nominal4* does not have the representation necessary to reach a reliable conclusion in respect to the performance of the models.

4. Large Scale Marine Protected Areas and Global Fishing Fleets

In this section we tackle a real world problem - how certain characteristics of Large Scale Marine Protected Areas (LSMPAs) and geographic-influenced factors impact fishing effort within and near a LSMPA. For such purpose, we perform a comprehensive study on predictive models trained with a data set with information about thirteen LSMPAs [6], using state-of-art interpretability methods as well as the accountability tools proposed in Section 3.

Marine Protected Areas (MPAs) have been designated around the world's oceans to enhance global marine protection and to counteract threats originated by overfishing, coastal development and climate change. LSMPAs encompass MPAs with an area over 100 000km^2 and should be *actively managed for protection across the entire geographic extent of the area* [25]. Established and planned future LSMPAs will soon constitute 95% of the global marine protected area [6], but their fairly recent existence calls for a reevaluation of some ecological and socio-economic factors as initially perceived for



t

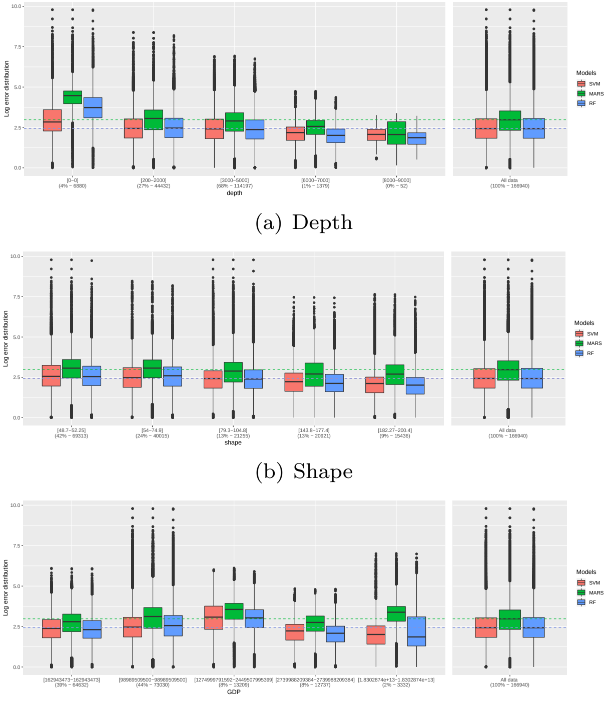
Figure 6: MEDP for feature *Attribute1* of data set *acceleration* using logarithmic scale.

smaller MPAs. In fact, despite the restriction or prohibition of fishing activities in these areas, there is a scarcity of information on how LSMPAs interact with surrounding fisheries. Thus, the object of this case study are these large-scale MPAs.

With this aim, we analysed how a variety of factors influence the fishing effort in thirteen LSMPAs established before 2015. This study investigates the area inside and around each MPA, evaluating regions within a radius of up to 500km from the border, allowing for an insight on the influence of the MPA in neighboring zones. Regarding the data set used (further detailed in [6]), the target variable we are interested is the fishing effort, measured by fishing hours, and the predictor variables in study are environmental (sea surface temperature and ocean productivity), physical (depth, distance to the MPA boundary, distance to high seas, area, shape, percentage of buffer in high seas and percentage of MPA bordering the high seas) and economic (age, GDP, enforcement, number of management zones, number of protection categories and percentage of no-take). The first five predictors mentioned are variable within each MPA, while the remaining parameters are fixed and relative to each particular LSMPA.

An exhaustive analysis of each individual LSMPA is extremely extensive to be fully represented here, so we will focus on the global analysis. However, these results can be consulted in the web page `https://github.com/inesareosa/MScThesis`.

With the aim of investigating how each characteristic and parameter of a MPA relates to the fishing effort, we initially compared the accuracy of 3 algorithms to choose the most adequate one: a Support Vector Machine, a Multivariate Adaptive Regression Spline (MARS) and a Random Forest, calculating the error estimates of each data instance using 10-fold CV. The parameters for each algorithm were tuned using the R package *performanceEstimation* [23].



(a) Depth

(b) Shape

(c) GDP

Figure 7: MEDP of features from Fishing Effort analysis

The median expected absolute error for the three algorithms was calculated, with the SVM and the RF presenting the best overall performance in terms of this metric (10.365 and 10.309 respectively), while the MARS had the worse performance, with a median error of 18.614. As stated previously, using a single metric for quantifying an overall performance might conceal some particularities in respect to certain values of the domain. Furthermore, the similar median expected error between the Random Forest and the Support Vector Machine demand further investigation for sensibly selecting the most adequate model. Hence, we employed MEDPs across all predictor values, using the quantiles of the values to select the ranges of the bins.

The MEDPs, plotted in Figure 7, showed that the estimated performance improves with the increase of *Depth* and *Shape*, being that for the former we can actually establish some differences between the SVM and the RF, since the SVM outperforms for cases when depth is of 0m, while the RF outperforms for values of depth superior to 6000m. The MEDP of feature *GDP* also shows some variability, with the SVM and the RF showing an improved estimated error distribution for countries with an higher GDP. It was concluded that for all other features the expected performance tends to behave similarly across the values of the domain. Therefore, we can infer that the model performance is not related to the values of these predictors. All MEDPs disclosed a consistent underperformance from the MARS model, independently of the operating domain.

In light of previous considerations, we can infer that MARS is clearly outperformed by the SVM and RF algorithms for this problem of fishing effort. Moreover, the differentiation between the usage of the SVM versus the RF depends on the values of the domain in which the end user would want to operate on. However, for further analysis we opted to utilise just the Random Forest since it has the better median expected absolute error, besides being the most commonly used algorithm in this field.

Up to this point the interaction between predictors was disregarded. The usage of PEPs allow for an analysis of the performance across the values of all the predictors simultaneously, thus enabling the inspection of possible interactions that might lead to performance degradation.

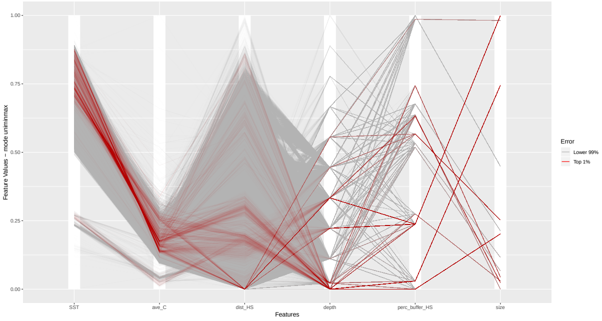


Figure 8: PEP for the RF that models all the 13 LSMPAs, for the 6 most important predictors

Figure 8 depicts the Parallel Error Plot for the 6 most important predictors (computed with R package *caret*) of the RF that models all LSMPAs. This plot shows that the top 1% of the errors are likely to occur in situations with high temperatures, low values of ocean productivity and extremely low to central depths, not having any particular visible relation with the distance to high seas, percentage of buffer in high seas and size.

Accountability methods do not provide the entire information required to fully understand the problem. Therefore, in a second phase, some selected state-of-the-art interpretability methods were applied to discover which factors influence fishing within and near LSMPAs. Local methods were not adopted since the main purpose is to obtain an overlook of the influence of a variety of factors behind fishing effort patterns, without any particular interest in explaining specific predictions.

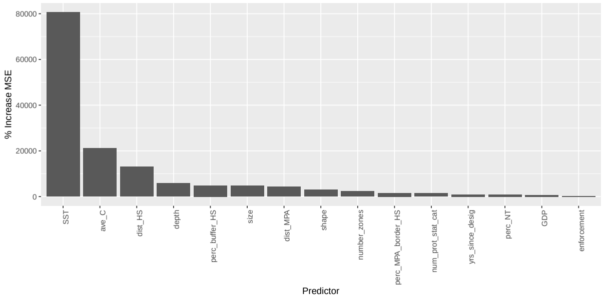
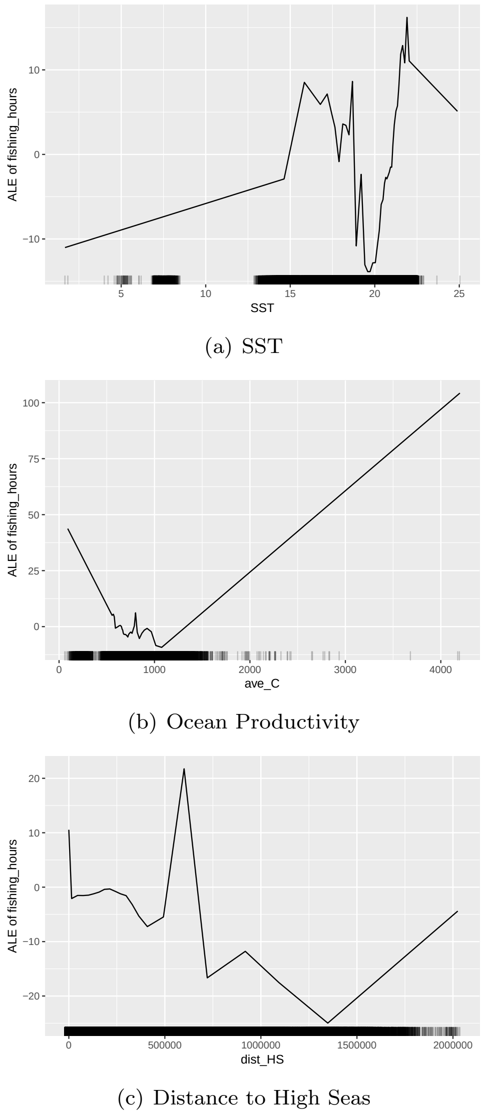


Figure 9: PFI results: % Increase in the Mean Squared Error for each feature of the Random Forest that models the LSMPAs

The Permutation Feature Importance (PFI) [7] enables the calculation of a score that selects the most important features for a given model, being model-specific for Random Forests. This computation was performed using the R package randomForest [16]. Figure 9 plots the results, showing that the primary drivers of overall fishing effort are environmental factors, particularly the *SST* and the *Ocean Productivity*. This was not unexpected since optimal temperatures lead to high concentration of plankton, which in turn increases the concentration of fish. We emphasize the role of the *Percentage of Buffer in the High Seas*, which is considered with this method as the most important MPA characteristic for determining the fishing effort.

To capture the influence of each predictor variable in the fishing effort patterns, we used ALE plots, which plot the estimated variation of the target value for each value of the predictor. These were calculated using the function *ale* from R package *iml* [17].

We concluded that most variables show clear influence on defining fishing effort patterns. In Figure 10 the ALE plots for the three features considered as most important are depicted. The highest fishing effort values are encountered in shallower waters, with temperatures between 15°C and 17°C or above 21°C, as well as for places with very low or very high ocean productivity (below approximately $875 mgC/m^2/day$ or above 1375 $mgC/m^2/day$).



(a) SST

(b) Ocean Productivity

(c) Distance to High Seas

Figure 10: ALE Plots of the fishing effort in relation to predictors of the Random Forest that models the LSMPAs

The fishing hours tendentially increase with the distance to the Marine Protected Area and show some variability with the distance to high seas, with greater effort in areas located in high seas. Regarding the parameters specific to each LSMPA, higher fishing effort apparently occurs for older and bigger MPAs with greater shape ratio, when designated by countries with a lower GDP, lower enforcement and fewer number of management zones but with stronger protection. Fishing hours tend to decrease the higher the percentage of buffer located in high seas, and slightly increase with the percentage of MPA bordering high seas, apart for cases with extremely high percentages, where the fishing effort is lower.

All the algorithms that evaluate the influence of predictor values in the model outcome ignore interactions between those same predictors, that can influence the model prediction. The H-Statistic is a metric that estimates the degree of interactions between features, in a scale from 0 to 1, with 0 reporting no interaction and 1 informing that the effect on a prediction arises solely from interaction. This value was obtained using R package *iml* [17].

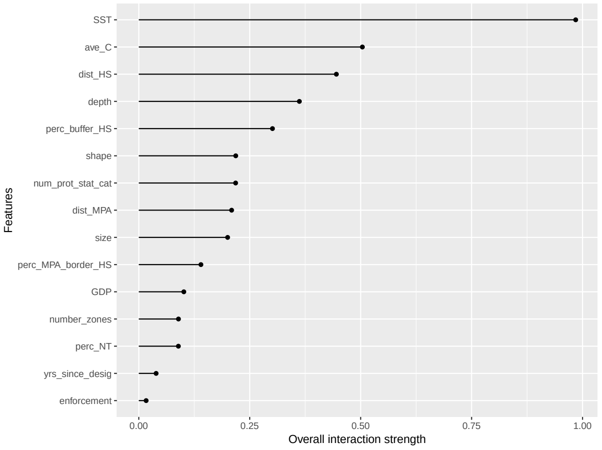


Figure 11: H-Statistic for the Random Forest that models all 13 LSMPAs together

Figure 11 plots the H-Statistic for the predictors of the Random Fores. The value for the *Sea Surface Temperature*, close to 1, shows that this feature impacts the outcome almost solely by means of interaction with other features. This is a curious remark if taking into consideration that this was considered as the most important feature. *Ocean Productivity*, *Distance to High Seas*, *Depth* and *Percentage of Buffer of High Seas* also present a relatively high H-Statistic.

5. Conclusions

The demand for transparency calls for methods that enable an insight on the functioning of black box models, to better understand them and consequently trust them. This paper describes a series of model agnostic visual tools designed to address the problem of explainability. In addition to interpret the model itself, it is crucial to anticipate the risks associated with trusting the models.

We describe a novel approach, in which we try explain the relation between the expected error and the values of the predictor variable. We suggested three new tools that increase the ability of end users to correctly access the risks behind predicting for a certain test case. We propose and evaluate the Error Dependence Plots (EDPs), which visualise the expected error distribution of a model against the values of a predictor variable. Next, we extend EDPs to a multivariate setting with Parallel Error Plots (PEPs). Later, a variant of EDPs to compare different models is presented - the MEDP. All tools, code and data used in this paper are available in `https://github.com/inesareosa/MScThesis`. We have presented some illustrative examples of these tools that showcase their utility in estimating the risk associated with using a certain algorithm.

Lastly, we investigated a case about the interaction of Large Scale Marine Protected Areas (LSMPAs) with fishing fleets. We provided a comparative analysis between three algorithms, in which we concluded that the SVM and the RF have a better performance than MARS to this problem. Later, the chosen algorithm, a RF, was further scrutinized in terms of performance and interpretability. This study lead to some interesting findings in terms of feature importance, relation of features with outcome and interactions between features.

As further work, we believe an interesting research direction could be related to the development of a tool that would ally interpretability and performance analysis. In what regards the fishing effort case study, we would like to inspect the model using local methods to determine reasons behind extremely high predicted fishing effort.

References

[1] A. Adadi and M. Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160, 2018.

[2] H. Akaike. *Information Theory and an Extension of the Maximum Likelihood Principle*, pages 199–213. Springer New York, New York, NY, 1973.

[3] T. Anderson and D. Darling. Asymptotic theory of certain goodness of fit criteria based on stochastic processes. *The Annals of Mathematical Statistics*, 23:193–212, 06 1952.

[4] D. Apley. Visualizing the effects of predictor variables in black box supervised learning models. 12 2016.

[5] J. Bi and K. P. Bennett. Regression error characteristic curves. In *Proc. of the 20th Int. Conf. on Machine Learning*, pages 43–50, 2003.

[6] K. Boerder, B. O'Leary, C. McOwen, E. Madin, D. M. McCauley, C. Jablonicky, L. T. Torgo, M. Dureuil, D. P. Tittensor, and B. Worm. Interactions between large marine protected areas and global fishing fleets *(under review)*.

[7] L. Breiman. Random forests. *Mach. Learn.*, 45(1):18–21, Oct. 2001.

[8] E. Dimitriadou, K. Hornik, F. Leisch, D. Meyer, and A. Weingessel. *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*, 2011.

[9] A. Fisher, C. Rudin, and F. Dominici. All models are wrong but many are useful: Variable importance for black-box, proprietary, or misspecified prediction models, using model class reliance, 2018.

[10] J. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29:1217–1222, 11 2000.

[11] B. Greenwell, B. Boehmke, J. Cunningham, and G. Developers. *gbm: Generalized Boosted Regression Models*, 2018.

[12] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. A survey of methods for explaining black box models. *ACM Comput. Surv.*, 51(5):93:1–93:42, Aug. 2018.

[13] R. J. Hyndman and A. B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, pages 679–688, 2006.

[14] A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(2):69–91, 1985.

[15] M. Kuhn. *caret: Classification and Regression Training*, 2019.

[16] A. Liaw, M. Wiener, L. Breiman, and A. Cutler. *randomForest: Breiman and Cutler's Random Forests for Classification and Regression*, 2018.

[17] C. Molnar. *iml: Interpretable Machine Learning*, 2019.

[18] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2017.

[19] F. Scholz and A. Zhu. *kSamples: K-Sample Rank Tests and their Combinations*, 2019.

[20] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, Mar. 1978.

[21] M. Shcherbakov, A. Brebels, N. Shcherbakova, A. Tyukov, T. Janovsky, and V. Kamaev. A survey of forecast error measures. *World Applied Sciences Journal*, 24:171–176, 01 2013.

[22] L. Torgo. Regression error characteristic surfaces. In *KDD'05: Proc. of the 11th ACM SIGKDD*, pages 697–702, 2005.

[23] L. Torgo. *performanceEstimation: An Infra-Structure for Performance Estimation of Predictive Models*, 2016.

[24] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S.* Springer, 4th edition, 2002.

[25] D. Wagner, A. Wihlem, A. Friedlander, A. Skeat, A. Sheppard, B. Bowen, C. Gaymar, G. Martin, I. Wright, J. Philibotte, J. Parks, J. Bosanquet, K. Aiona, J. brider, K. Morishige, L. Wright-Koteka, N. Lewis, N. Brownie, R. Kosaki, and Z. Basher. 02 2013.