arXiv:2007.06894v1 [stat.ML] 14 Jul 2020

Model-Agnostic Interpretable and Data-driven suRRogates suited for highly regulated industries

Roel Henckaerts ROEL.HENCKAERTS@KULEUVEN.BE
Department of Accounting, Finance and Insurance
KU Leuven, Leuven, Belgium

Katrien Antonio KATRIEN.ANTONIO@KULEUVEN.BE
Department of Accounting, Finance and Insurance
KU Leuven, Leuven, Belgium

Marie-Pier Côté MARIE-PIER.COTE@ACT.ULAVAL.CA
École d'actuariat
Université Laval, Québec, Canada

Abstract

Highly regulated industries, like banking and insurance, ask for transparent decision-making algorithms. At the same time, competitive markets push for sophisticated black box models. We therefore present a procedure to develop a Model-Agnostic Interpretable Data-driven suRRogate, suited for structured tabular data. Insights are extracted from a black box via partial dependence effects. These are used to group feature values, resulting in a segmentation of the feature space with automatic feature selection. A transparent generalized linear model (GLM) is fit to the features in categorical format and their relevant interactions. We demonstrate our R package `maidrr` with a case study on general insurance claim frequency modeling for six public datasets. Our maidrr GLM closely approximates a gradient boosting machine (GBM) and outperforms both a linear and tree surrogate as benchmarks.

Keywords: Business Practice & Compliance, GLM, Insurance, Segmentation, XAI

1. Introduction

The big data revolution opens the door to complex artificial intelligence (AI) technology in search for top performance. At the same time, there is growing public awareness for the issues of interpretability, explainability and fairness (O'Neil, 2016). The General Data Protection Regulation (GDPR, 2016) establishes a regime of "algorithmic accountability" and "the right to an explanation" of decision-making algorithms. Highly regulated industries require an extensive review of algorithms by supervisory authorities and demand transparent communication to customers on the reasoning behind an algorithm's decisions. Examples from the financial sector are the key information documents (KIDs) for packaged retail and insurance-based investment products (PRIIPs, 2014), detailed motivations for credit actions under the Equal Credit Opportunity Act (ECOA, 1974) and filing requirements for general insurance rates to the National Association of Insurance Commissioners (NAIC, 2012).

Algorithmic comprehensibility hinders AI implementations in business practice due to regulatory compliance (Arrieta et al., 2020). An explainable AI (XAI) algorithm enables human users to understand, trust and manage its decisions (Gunning, 2017). There is a clear distinction between explainability via interpretation techniques *after the event* and interpretability or transparency *by design* (Guidotti et al., 2018). On the one hand, a wide

range of interpretation techniques are available to aid users in the explainability of opaque models and their predictions (Biecek, 2018). On the other hand, decision trees, rules and linear models are considered to be transparent, meaning they are easily comprehensible for human users. Linear models are interpretable by design as the contribution (sign and strength) of feature x_j to the prediction target y is directly observable from the model coefficient β_j (Doran et al., 2017). Furthermore, the output is simply visualized in a decision table, see Figure 1. Huysmans et al. (2011) perform a user study on the comprehensibility of several representation formats and show that decision tables outperform trees and rules with respect to accuracy, response time, answer confidence and ease of use.

General formulation of a linear model:
$\mathbb{E}[y] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$

Return (%) based on asset class and investment term:
$\mathbb{E}[\texttt{return}] = 2 + 4\,\texttt{asset}_{stock} + 3\,\texttt{term}_{long}$

Asset	Term	$\mathbb{E}$[return]
bond	short	2%
bond	long	5%
stock	short	6%
stock	long	9%

Figure 1: An example of a linear model (left) and the corresponding decision table (right).

This paper presents a procedure to develop a transparent model based on knowledge extracted from a black box via interpretation techniques. The goal is to obtain a global surrogate that inherits the strengths of a sophisticated black box algorithm, delivered in a format that is easier to understand, manage and implement thanks to the reduced complexity. An automatic procedure resulting in a high degree of model transparency can boost AI applications in the business, especially in highly regulated sectors as banking and insurance.

We put forward the following three desirable properties for a simplification procedure. Firstly, a *model-agnostic* procedure is preferred due to the ever increasing variety of black box algorithms. We rely on partial dependence (PD) effects to extract insights from the black box, thereby covering a vast amount of different model types (Friedman, 2001). Secondly, the surrogate should be *interpretable* such that human users can easily comprehend and use the model. We choose to use generalized linear models (GLMs), formulated by Nelder and Wedderburn (1972). This is a versatile model class, widely used in the insurance industry, which still allows to visualize the output as a decision table. Thirdly, a *data-driven* procedure avoids the need for quick-and-dirty model choices. We employ a cross-validation scheme to fully automate the transformation from a black box to a transparent model.

We propose maidrr: a procedure to obtain a Model-Agnostic Interpretable Data-driven suRRogate for a black box developed on structured tabular data. Our approach is related to the ideas of model compression (Bucilă et al., 2006), mimic learning (Ba and Caruana, 2014) and distillation (Hinton et al., 2015). These papers aim to transfer knowledge from a large/slow model into a compact/fast approximation, which can easily be deployed in environments with stringent space and time requirements. We see two major differences between our work and the aforementioned papers. Firstly, these papers learn the underlying structure of the complex model by using its output as (soft) labels for training the simpler model. We do not use the predicted output of the complex model to develop the simpler

model, but instead use the extracted knowledge to perform smart feature engineering on the original training data. Secondly, in the aforementioned papers the resulting simple models are shallow neural nets which still need interpretation techniques to become explainable. We deliver a transparent GLM which is directly comprehensible for human users.

The rest of this paper is structured as follows. Section 2 details the maidrr methodology. Section 3 shows an application to insurance claim frequency modeling, demonstrating how maidrr outperforms a linear and tree benchmark surrogate. Section 4 concludes this paper.

2. Methodology

We first give an overview of the process behind maidrr, schematized in Figure 2. Afterward, we describe each step in details. The starting point is a black box that we want to replace with a simpler and more comprehensible surrogate. We extract insights from the black box in the form of partial dependence (PD) effects for all features involved. These PD effects, detailing the relation between a feature and the target, are used to group values/levels within a feature via dynamic programming (DP). A slightly different grouping approach is used for different types of features. For continuous/ordinal features, only adjacent values should be binned together, whereas any two levels within a nominal feature can be clustered. The binning/clustering via DP leads to an optimal and reproducible grouping of feature levels, resulting in a full segmentation of the feature space. A generalized linear model (GLM) is fit to the segmented data with all features in a categorical format and their relevant interactions, thereby producing an interpretable model as end product.

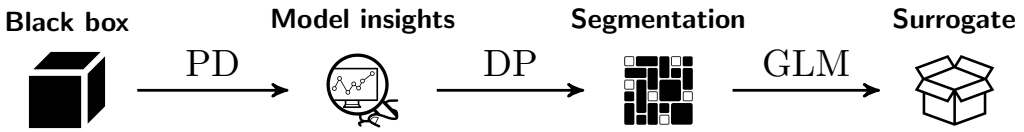


Figure 2: The maidrr process to replace a black box algorithm with a transparent GLM.

Black box As a starting point, any black box model giving a prediction function $f_{\text{pred}}(\boldsymbol{x})$ for features $\boldsymbol{x} \in \mathbb{R}^p$ can be used. This property makes maidrr a model-agnostic procedure.

Model insights A univariate partial dependence (PD) captures the marginal relation between a feature and the model predictions (Friedman, 2001). The prediction function f_{pred} is evaluated in specific values of the feature of interest x_j, for given $j \in \{1, \ldots, p\}$, while being averaged over n observed values of the other features $\boldsymbol{x}^i_{-j}$ for observation $i \in \{1, \ldots, n\}$:

$$\bar{f}_j(x_j) = \frac{1}{n} \sum_{i=1}^{n} f_{\text{pred}}(x_j, \boldsymbol{x}^i_{-j}). \tag{1}$$

PD effects measure global model behavior due to the averaging over n training observations. The PD effect $\bar{f}_j$ is used to group values/levels within feature x_j, since values of feature x_j with a similar PD effect show a similar relation to the prediction target. By grouping these values together, we reduce the complexity of the feature with a limited loss of information. For feature x_j, let m_j denote the unique number of observed values in the data and let $x_{j,q}$

denote its qth value for $q \in \{1, \ldots, m_j\}$. We then define $z_{j,q} = \bar{f}_j(x_{j,q})$ as the PD effect of feature x_j evaluated in $x_{j,q}$. The goal is now to group the values $x_{j,q}$ in k_j groups based on the calculated PD effects $z_{j,q}$. Since we group each feature x_j separately, this can be seen as a one-dimensional clustering problem of $z_{j,q}$ for $q \in \{1, \ldots, m_j\}$.

Segmentation Wang and Song (2011) developed a dynamic programming (DP) algorithm for optimal and reproducible one-dimensional clustering problems. Elements of an m_j-dimensional input vector are assigned to k_j clusters by minimizing the within-cluster sum of squares, that is, the sum of squared distances from each element to its corresponding cluster mean. This follows the same spirit as the classical K-means algorithm (MacQueen, 1967), but the DP algorithm guarantees reproducible and optimal groupings by progressively solving the sub-problem of clustering u elements in v clusters with $1 \leq u \leq m_j$ and $1 \leq v \leq k_j$. This algorithm is implemented in the `R` package `Ckmeans.1d.dp` (Song, 2019) and allows for the inclusion of adjacency constraints in the clustering problem. For continuous/ordinal features, we impose such constraints, since we only want to group adjacent values. Nominal features are clustered without adjacency constraints such that any two levels can be grouped. The DP algorithm requires the specification of the number of groups k_j for feature x_j. In theory, we could perform a p-dimensional grid search to find the optimal number of groups for each feature x_j with $j \in \{1, \ldots, p\}$. However, this would cause the computation time to grow exponentially with p, harming maidrr's scalability.

Optimal number of groups After grouping feature x_j in k_j groups, let $\tilde{z}_{j,q}$ represent the average PD effect for the group to which $x_{j,q}$ belongs. We define a penalized loss function, which is to be minimized to find the optimal k_j from a range of values, as follows:

$$\frac{1}{m_j} \sum_{q=1}^{m_j} \left(z_{j,q} - \tilde{z}_{j,q}\right)^2 + \lambda \log(k_j). \tag{2}$$

The first part of this loss function measures how well the PD effect is approximated by the grouped variant as a mean squared error (MSE) over all unique values of feature x_j. The second part of Eq. (2) measures the complexity by means of the common logarithm of the number of groups k_j. The penalty parameter λ acts as a bias-variance trade-off. A low (high) value of λ allows for many (few) groups, resulting in an accurate (coarse) approximation of the PD. Note that λ does not depend on j in Eq. (2), which is adequate because the PD effects reside on the same scale, namely the scale of the predictions, see Eq. (1). The original p-dimensional tuning problem in this way reduces to be one-dimensional over λ.

Surrogate Given a λ value, we minimize Eq. (2) for each of the features x_j, resulting in a full segmentation of the feature space. After this step of feature engineering based on black box insights, we fit a transparent model to the original target and features in a categorical format. Generalized linear models (GLMs) allow for the specification of a diverse set of target distributions (Nelder and Wedderburn, 1972). This facilitates the application of maidrr to classification tasks and many types of regression problems, for example linear, Poisson and Gamma regression. See Appendix A for details on the GLM formulation. GLMs with only categorical features lead to fixed-size decision tables, see Appendix B. GLMs with many features remain transparent, fileable in a tabular format and easy to use by business intermediaries, so the complexity of the GLM is not a concern.

Feature interactions So far, we focused on grouping features via their marginal PDs. However, interactions between features can play a major role in explaining the data. We first find a set of relevant interactions by considering their strength as measured via the H-statistic of Friedman and Popescu (2008). Then, the interaction between features x_a and x_b is captured by subtracting both one-dimensional PDs from the two-dimensional PD:

$$\bar{f}_{a,b}(x_a, x_b) = \frac{1}{n}\sum_{i=1}^{n} f_{\text{pred}}(x_a, x_b, \boldsymbol{x}^i_{-a,-b}) - \frac{1}{n}\sum_{i=1}^{n}\sum_{\ell\in\{a,b\}} f_{\text{pred}}(x_\ell, \boldsymbol{x}^i_{-\ell}). \tag{3}$$

The interaction effect between features x_a and x_b is grouped by applying the DP algorithm to cluster similar $\bar{f}_{a,b}(x_a, x_b)$ values. We do not impose adjacency constraints as interactions represent a correction on top of the marginal effects and we prefer to allow for maximum flexibility. Given a value of λ, we determine the number of groups k_{ab} by minimizing the equivalent of Eq. (2) obtained by computing the first term with Eq. (3).

Tuning strategy Algorithm 1 details the maidrr procedure, where two hyperparameters need tuning: λ_{marg} and λ_{intr}. A distinct value of λ is advised for marginal and interaction effects respectively, as the PDs in Eq. (1) and (3) reside on different scales. Marginal PDs are centered around the average target prediction, whereas interaction PDs are around zero. We tune the λ's via K-fold cross-validation by iterating over a grid of λ values and choosing the optimal value that minimizes the error of the surrogate GLM predictions. This error is computed with regards to the original data and not the black box predictions, resulting in a data-driven procedure. The tuning can be performed in two stages, first for λ_{marg} and next for λ_{intr}, thereby avoiding a two-dimensional grid search and saving computation time. Automatic feature selection is enabled as feature x_j is excluded from the surrogate when $k_j = 1$. The hyperparameter h selects a set of relevant interactions by means of a cut-off on the realized values of the H-statistic, thereby excluding unimportant interactions upfront.

Algorithm 1 maidrr

Input: data, f_{pred}, λ_{marg}, λ_{intr} and h
Output: surrogate GLM
for $j = 1$ **to** p **do**
  calculate the PD effect $\bar{f}_j$ via Eq. (1)
  apply the DP algorithm to feature x_j with $k_j = \arg\min$ Eq. (2) for $\lambda = \lambda_{\text{marg}}$
end for
Set $I = \{(l, m) \mid H(x_l, x_m) \geq h\}$
for all (a, b) **in** I **do**
  calculate the PD effect $\bar{f}_{a,b}$ via Eq. (3)
  apply the DP algorithm to interaction (x_a, x_b) with $k_{ab} = \arg\min$ Eq. (2) for $\lambda = \lambda_{\text{intr}}$
end for
fit GLM with selected features and interactions in categorical format

3. Case study for the insurance industry

Insurers are required by law to document their pricing or rating model to the regulator, creating a clear need for a fully transparent format. A crucial part of ratemaking is the accurate modeling of the number of claims reported by a policyholder. We therefore apply maidrr to a general insurance claim frequency prediction problem. Section 3.1 introduces the model setting and the datasets. Section 3.2 details the model construction for the black box and the GLM surrogate obtained via maidrr. Section 3.3 evaluates the approximation performance of the GLM with respect to the black box against two benchmark surrogates.

3.1 Claim frequency modeling with insurance data

We analyze six motor third party liability (MTPL) insurance portfolios, which are available in the R packages `maidrr` (Henckaerts, 2020) or `CASdatasets` (Dutang and Charpentier, 2019). All datasets contain an MTPL portfolio followed over a period of one year, with the amount of policyholders (n) and the number of features (p) detailed in Table 1. Each dataset holds a collection of different types of risk features, for example the age of the policyholder (continuous), the region of residence (nominal) and the type of insurance coverage (ordinal).

	`ausprivauto`	`bemtpl`	`freMPL`	`freMTPL`	`norauto`	`pricingame`
n	67,856	163,210	137,254	677,925	183,999	99,859
p	5	10	9	8	4	19

Table 1: Overview of the number of policyholders (n) and features (p) in the datasets.

We model the number of claims filed during a given period of exposure-to-risk, defined as the fraction of the year for which the policyholder was covered by the insurance policy. Exposure is vital information, as filing one claim during a single month of coverage represents a higher risk than filing one claim during a full year. Table 2 shows the distribution of the number of claims in the portfolios. Most policyholders do not file a claim, some file one claim and a small portion files two or more claims. Such count data is often modeled via Poisson regression, a specific form of GLM with a Poisson assumption for the target y and a logarithmic link function. In this setting, industry standard is to incorporate the logarithm of exposure t via an offset term: $\ln(\mathbb{E}[y]) = \ln(t) + \beta_0 + \sum_j \beta_j x_j$. This leads to $\mathbb{E}[y] = t \times \exp(\beta_0 + \sum_j \beta_j x_j)$, that is, predictions which are proportional to exposure.

	0	1	2	3	4	5	6
`ausprivauto`	63,232	4333	271	18	2	0	0
`bemtpl`	144,936	16,539	1554	162	17	2	0
`freMPL`	106,577	26,068	4097	448	62	2	0
`freMTPL`	643,874	32,175	1784	82	7	2	1
`norauto`	175,555	8131	298	15	0	0	0
`pricingame`	87,213	11,232	1262	134	16	1	1

Table 2: Distribution of the number of claims in the portfolios.

3.2 Finding a transparent model by opening the black box

This section elaborates on the construction of the different models: a GBM black box in Section 3.2.1 and our proposed GLM surrogate obtained via maidrr in Section 3.2.2.

3.2.1 GBM as black box

We opt for a gradient boosting machine or GBM (Friedman, 2001) as the black box to start from. More specifically, we make use of stochastic gradient boosting (Friedman, 2002) as implemented in the R package `gbm` (Greenwell et al., 2019). This choice is based on the performance of GBMs in related work (Henckaerts et al., 2020). Due to the model-agnostic set up of maidrr, any model can be used as input, including deep neural networks.

We tune the number of trees T in the GBM via 5-fold cross-validation, see Table 3. Other hyperparameters are fixed to a sensible value. Following Hastie et al. (2009, Section 10.11), we use decision trees of depth two, which are able to model up to third-order interactions. Each tree is built on randomly sampled data of size $0.75n$ and the learning rate is set to 0.01. To take into account the distributional characteristics of the count data, we use the Poisson deviance as loss function in the GBM tuning process. The Poisson deviance is defined as:

$$D^{\text{Poi}}\left\{y, f_{\text{pred}}(\boldsymbol{x})\right\} = \frac{2}{n}\sum_{i=1}^{n}\left[y_i \times \ln\left\{\frac{y_i}{f_{\text{pred}}(\boldsymbol{x}_i)}\right\} - \left\{y_i - f_{\text{pred}}(\boldsymbol{x}_i)\right\}\right]. \tag{4}$$

	`ausprivauto`	`bemtpl`	`freMPL`	`freMTPL`	`norauto`	`pricingame`
T	474	3214	1377	3216	793	1198

Table 3: Overview of the optimal number of trees (T) in the GBM for the different datasets.

3.2.2 GLM surrogate via maidrr

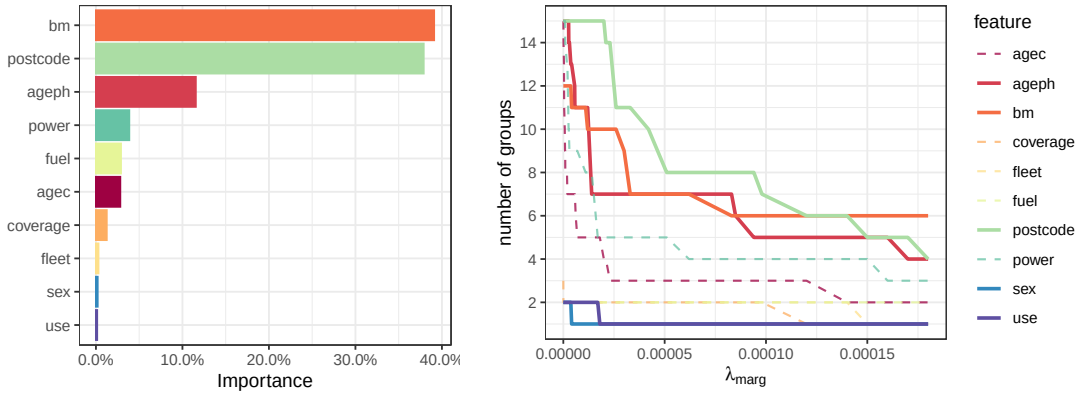
We build a surrogate GLM to approximate the optimal GBM for each dataset. The function `maidrr::autotune` (Henckaerts, 2020) implements a tuning procedure for Algorithm 1.

Algorithm 1 requires three input parameters: λ_{marg}, λ_{intr} and h. The λ values determine the granularity of the resulting segmentation and GLM. We define a search grid for both λ's, ranging from 10^{-9} to 1. To find the optimal values, shown in Table 4, we perform 5-fold cross-validation on the GLM with the Poisson deviance from Eq. (4) as loss function. This tuning is done in two stages. First, a grid search over λ_{marg} finds the optimal GLM with only marginal effects. Then, a grid search over λ_{intr} determines which interactions should be included in that GLM. This requires two one-dimensional grid searches instead of one two-dimensional search. The value of h determines the set of interactions that are considered for inclusion in the GLM by excluding meaningless interactions with a low H-statistic. This value is calculated automatically to consider the minimal set of interactions for which the empirical distribution function of the H-statistic exceeds 50%. The intent is to take into account the most important interactions while still keeping the GLM simple.

	ausprivauto	bemtpl	freMPL	freMTPL	norauto	pricingame
λ_{marg}	8.1×10^{-5}	5.6×10^{-6}	2.9×10^{-5}	1.3×10^{-6}	1.1×10^{-5}	2.1×10^{-6}
λ_{intr}	2.5×10^{-4}	1.5×10^{-5}	1.4×10^{-2}	3.2×10^{-6}	4.6×10^{-5}	3.3×10^{-5}

Table 4: Overview of the optimal λ_{marg} and λ_{intr} values for the different datasets.

We put focus on results for the `bemtpl` portfolio. Figure 3(a) shows feature importance scores according to the GBM. Figure 3(b) shows the number of groups for each feature in function of λ_{marg}, where important features retain a higher number of groups for increasing values of λ_{marg}. This confirms that maidrr puts the levels of uninformative features in one group, as such excluding them from the GLM and performing automatic feature selection.

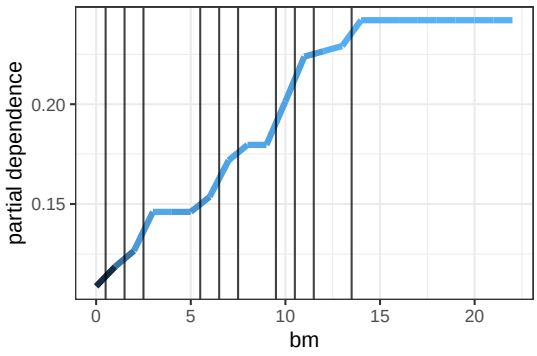


(a) Feature importance in the optimal GBM

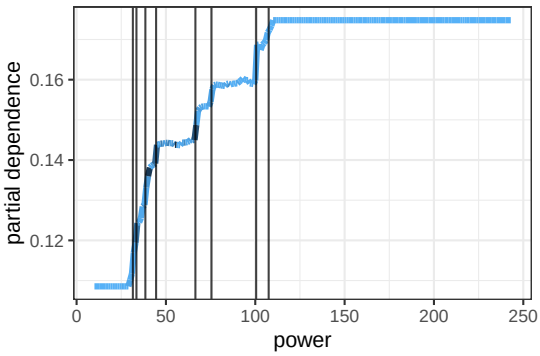
(b) Number of groups in function of λ_{marg}

Figure 3: Illustration of the automatic feature selection process in maidrr for `bemtpl`.

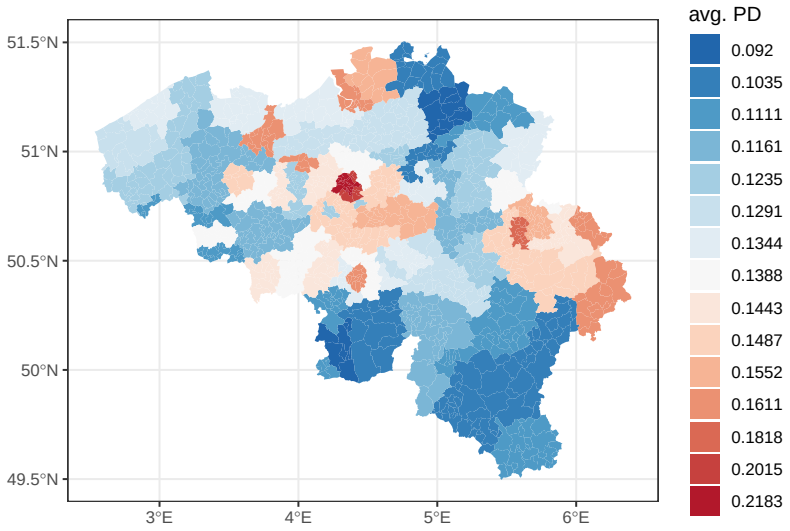
Figure 4 displays the resulting segmentation for selected features in the `bemtpl` portfolio. Figures 4(a) and 4(b) show the GBM PD for the bonus-malus level and power of the vehicle, where darker blue indicates a higher observation count in the portfolio. Both features have an increasing effect on the claim frequency and are grouped into 11 and 9 bins respectively, indicated by the vertical lines. The bins are wide wherever the effect is quite stable and narrow where the effect is steeper. Figure 4(c) shows the postal code groups on the map of Belgium. The initial 80 levels are grouped in 15 clusters, indicated by the different colors. The capital Brussels in the center of Belgium (red colored), together with other big cities (orange colored), are risky due to heavy traffic in those densely populated areas. The rural regions in the northeast and south of Belgium are less risky. The plotting characters in Figure 4(d), where size is proportional to total exposure, show how the limited and extensive material damage covers (TPL+ and TPL++) are grouped together due to their similar PD effect. Figure 4(e) shows the six groups, indicated by different shades of blue, for the interaction between the bonus-malus level and the vehicle power. This effect is considered to be a correction on top of the marginal effects in Figures 4(a) and 4(b). Observations with extreme values for both the bonus-malus level and vehicle power receive an extra risk penalty, while a combination of high/low values obtains a negative risk correction.



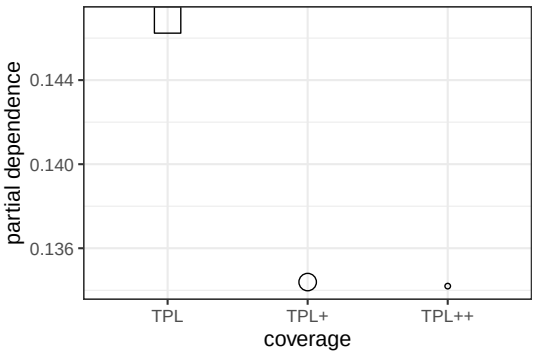
(a) Level in the bonus-malus scale



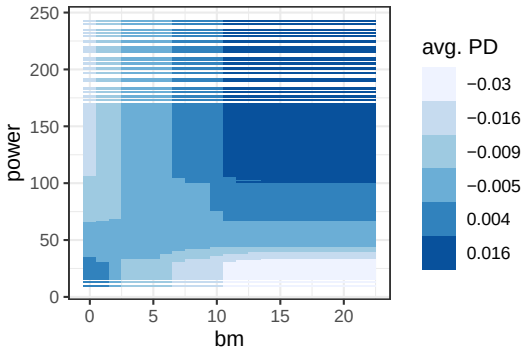
(b) Power of the vehicle



(c) Region of residence based on the postal code



(d) Type of coverage



(e) Bonus-malus level and vehicle power

Figure 4: Segmentation for several features in the `bemtpl` portfolio. Groups are indicated by vertical lines in (a) and (b), by colors in (c) and (e) and by plotting characters in (d). In (a), (b) and (d) we show the PD effect for each feature value/level, while in (c) and (e), we show the average PD effect per group.

3.3 Evaluation of the GLM surrogate

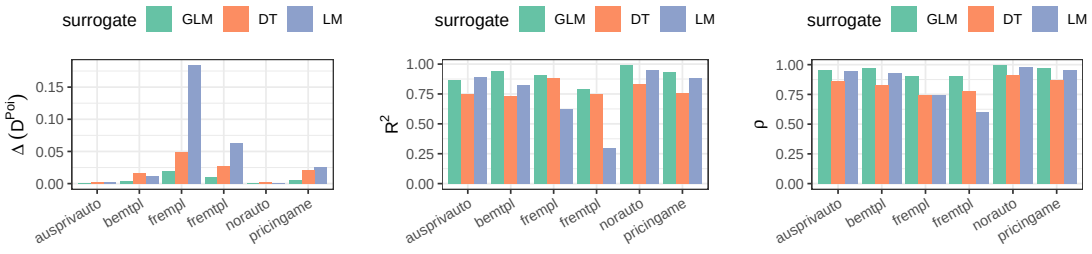
This section investigates how closely the maidrr GLM approximates the GBM black box with respect to predictions (Section 3.3.1) and explanations (Section 3.3.2). We benchmark our GLM against two transparent surrogates: a decision tree (DT) and linear model (LM). Both are fit with the original data as features and the GBM predictions as target (Molnar, 2020). We restrict the maximum tree depth to four to keep the result comprehensible.

3.3.1 Accuracy of the predictions

Figure 5 compares GLM (green), DT (orange) and LM (blue) surrogate predictions against the GBM for all datasets. We measure prediction accuracy for all models via the Poisson deviance from Eq. (4). With f_{surro} and f_{gbm} the surrogate and GBM prediction function, we assess the accuracy loss via relative differences as $D\{y, f_{\text{surro}}(\boldsymbol{x})\}/D\{y, f_{\text{gbm}}(\boldsymbol{x})\} - 1$. Figure 5(a) shows that the GLM outperforms the others on each dataset, with the averaged relative difference over all datasets equal to 0.7%, 2% and 4.8% for the GLM, DT and LM. We use the R^2 measure to quantify how accurately a surrogate is able to imitate or mimic the GBM's behavior. With μ_{gbm} the mean GBM prediction, the R^2 is defined as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \{f_{\text{surro}}(\boldsymbol{x}_i) - f_{\text{gbm}}(\boldsymbol{x}_i)\}^2}{\sum_{i=1}^{n} \{f_{\text{gbm}}(\boldsymbol{x}_i) - \mu_{\text{gbm}}\}^2}.$$

The $R^2 \in [0, 1]$ represents the percentage of variance captured by the surrogate model. Figure 5(b) shows that the GLM is ranked first in five datasets and second in the remaining one. The average R^2 for the GLM, DT and LM over all datasets equals 90.5%, 78.4% and 74.6%. We also compute the average of Pearson's linear and Spearman's rank correlation coefficients ρ between the GBM and surrogate predictions. Figure 5(c) shows that the GLM obtains first place in all datasets, while the average ρ for the GLM, DT and LM over all datasets equals 95.1%, 83.4% and 85.8%. We average Pearson's and Spearman's ρ to consolidate both types of correlation in one number, but the results also hold for each coefficient separately. In general, our GLM constructed with maidrr outperforms the benchmark DT and LM surrogates when it comes to prediction accuracy and mimicking the GBM.



(a) Relative difference in D^{Poi} (b) R^2 measure (c) Average ρ measure

Figure 5: Accuracy of the GLM (green), DT (orange) and LM (blue) surrogate predictions against the GBM: relative D^{Poi} difference (a), R^2 measure (b) and average ρ (c).

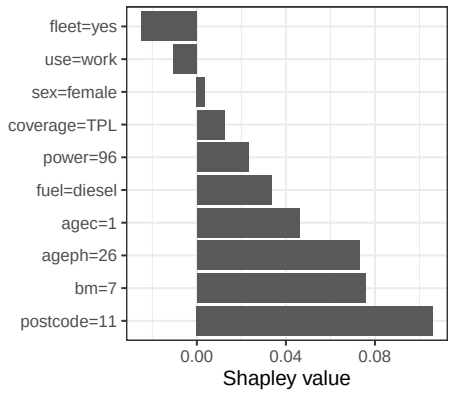
3.3.2 Similarity of the explanations

We now focus on explaining the predictions obtained with the GBM black box and our maidrr GLM surrogate, investigating the similarities or differences in these explanations. To explain the GBM we use Shapley (1953) values, with the efficient implementation of Štrumbelj and Kononenko (2010, 2014) available in the `R` package `iml` (Molnar et al., 2018). Explaining a GLM does not require any additional tools, since we can rely directly on the fitted GLM coefficients for this task. We focus on the `bemtpl` dataset and explain predictions for the three artificial instances listed in Table 5. Based on the GBM and GLM predictions, these instances represent a high/medium/low risk profile. The big question that automatically arises is: *How do the features influence the riskiness of each individual?*

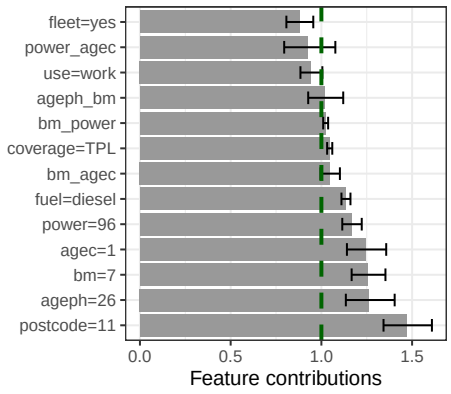
	high risk	medium risk	low risk
`bm`	7	4	1
`postcode`	11	91	55
`ageph`	26	45	66
`power`	96	66	26
`fuel`	diesel	gasoline	gasoline
`agec`	1	8	15
`coverage`	TPL	TPL+	TPL++
`fleet`	yes	no	no
`sex`	female	male	male
`use`	work	private	private
GBM	0.3035	0.1411	0.0609
GLM	0.4380	0.1330	0.0510

Table 5: Artificial instances for which we explain the GBM and GLM predictions.

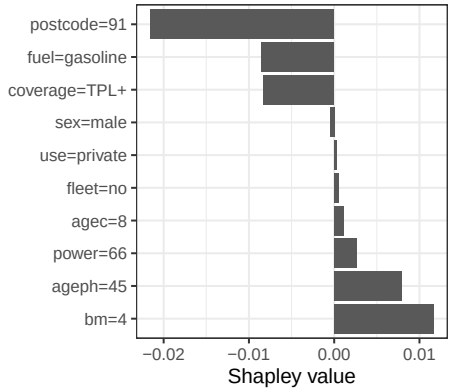
Figures 6(a), 6(c) and 6(e) show the Shapley values for each feature's contribution to the GBM prediction. The sum of these values equals the difference between the instance prediction, shown in Table 5, and the average GBM prediction of 0.1417. The presence of mainly positive (negative) values, see Figure 6(a) and 6(e), thus represents a high (low) risk profile. Figures 6(b), 6(d) and 6(f) explain the GLM predictions via the fitted β coefficients. Each feature's contribution is shown, similar to Shapley values, but with three important distinctions. First, in the GLM it is possible to split the contribution over marginal effects and interactions with other features. Second, 95% confidence intervals indicate the uncertainty associated with each contribution. Third, the contribution is shown on the response scale after taking the inverse link function: $\exp(\beta_j)$ for x_j. In the resulting rating model, these contributions are multiplied with the baseline, as captured by the intercept via $\exp(\beta_0)$. The green dashed line indicates the point of "no contribution" at $\exp(0)$. An insurance rate is determined by the product of claim frequency and severity, such that the contribution in the frequency GLM can be directly interpreted as a percentage premium/discount on the price. This representation boosts the ability to understand the feature contributions on the scale of interest and allows for manual intervention when deploying the model in practice.



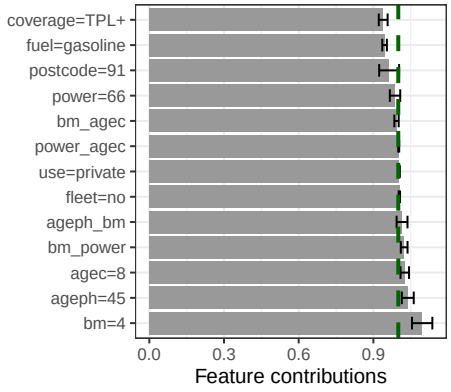
(a) GBM: high risk



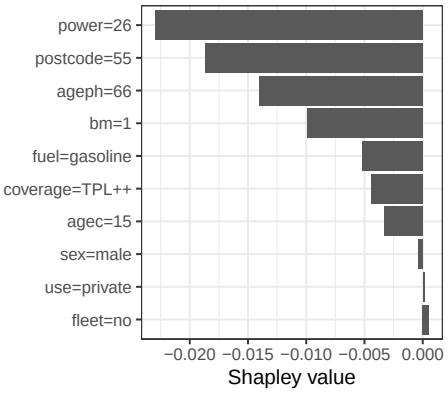
(b) GLM: high risk



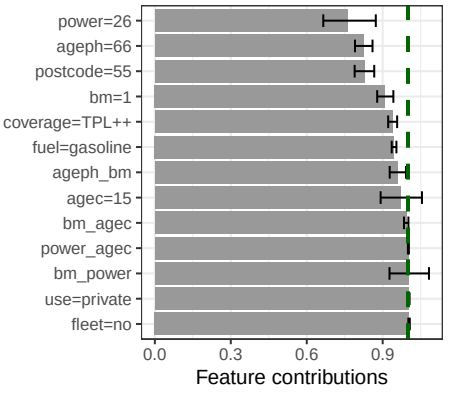
(c) GBM: medium risk



(d) GLM: medium risk



(e) GBM: low risk



(f) GLM: low risk

Figure 6: Explanations for the instance predictions from Table 5 in the GBM via Shapley values (left) and the GLM via the fitted β coefficients (right).

The GBM and GLM explanations are very similar, see Figures 6(a) and 6(b) for example. The high risk of this profile stems from a residence in Brussels, young age, high bonus-malus level and new high-powered diesel vehicle. Driving a fleet vehicle for work reduces the risk slightly. The GLM shows no contribution from gender as this feature is not selected by maidrr, while it is negligibly small in the GBM. The interaction between the power and age of the vehicle puts a negative correction on both positive marginal effects, but the confidence interval in the GLM displays the uncertainty coming with this effect. Living in Brussels increases the frequency, and thus the price, by approximately 50% in the technical analysis performed with this dataset. One can assess the fairness of this penalty, possibly followed by a manual adjustment to intervene in the decision-making process via expert judgment.

4. Conclusions

To accommodate the growing interest in explainability and transparency, we present maidrr: a procedure to develop a Model-Agnostic Interpretable Data-driven suRRogate. We apply maidrr to six real-life general insurance portfolios for claim frequency prediction, thereby focusing on a highly relevant count regression problem, which is not often dealt with in classical machine learning. We show that maidrr results in a transparent GLM which mimics the behavior of a black box GBM closely, outperforming two benchmark surrogates. In the process, maidrr automatically creates a complete segmentation of the feature space.

Explanations from the maidrr GLM only depend on the fitted coefficients, which are easily observable. This gives some important advantages to maidrr with respect to the following XAI goals (see Arrieta et al., 2020, Table 1). 1) *Trustworthiness:* a GLM with only categorical features always acts as intended since all the possible working regimes can be listed in a decision table of fixed size. 2) *Accessibility/Interactivity:* manual post-processing of the model becomes very easy and intuitive by tweaking the GLM coefficients. This allows users to intervene and be more involved in the development and improvement of the model. 3) *Fairness:* the clear influence of each feature allows for an ethical analysis of the model, which becomes especially important when a model's decisions influences people's lives. In our insurance setting, it is important that every policyholder receives a fair insurance quote. The direct interpretation of the feature contributions as a penalty/discount to the baseline tariff further aids this cause. 4) *Confidence:* the uncertainty of the contributions is quantifiable via confidence intervals such that the model's robustness, stability and reliability can be assessed. 5) *Informativeness:* contributions are split across marginal effects and interactions of features, thereby increasing the amount of information available to the user on the underlying decision of the model. We believe that these attributes make maidrr a viable alternative in any situation where a competitive, yet transparent model is needed.

Acknowledgments

This research is supported by the Research Foundation Flanders (SB grant 1S06018N) and by the Natural Sciences and Engineering Research Council of Canada (RGPIN-2019-04190). Furthermore, Katrien Antonio acknowledges financial support from the Ageas Research Chair at KU Leuven and from KU Leuven's research council (COMPACT C24/15/001).

Appendix A. GLM formulation

A GLM allows any distribution from the exponential family for the target of interest y. This includes, among others, the normal, Bernoulli, Poisson and gamma distribution, making GLMs a versatile modeling tool. Denoting by η the linear predictor and $g(\cdot)$ the link function, the structure of a GLM with all features $\boldsymbol{x}$ in a categorical format is as follows:

$$\eta = g(\mathbb{E}[y]) = \boldsymbol{\ell}^\top \boldsymbol{\beta} = \beta_0 + \sum_{j=1}^{d} \beta_j \ell_j.$$

The $d+1$ dimensional vector $\boldsymbol{\ell}$ contains a 1 for the intercept β_0 together with d dummy variables $\ell_j \in \{0, 1\}$. A categorical feature x with m levels contains a reference level which is captured by the intercept. The other $m-1$ levels are coded via dummy variables to model the differences between those levels and the reference level, captured by the coefficients β_j.

Appendix B. GLM in a tabular format

Table 6 shows part of the decision table for the `norauto` dataset. The three other parts for *Male = Yes & Young = No* and *Male = No & Young = Yes/No* are not shown for space reasons. Such a tabular model is easy to maintain in a spreadsheet with responsive filters.

	Male	Young	DistLimit	GeoRegion	GLM prediction (%)
1	Yes	Yes	8000 km	Low-	*3.49*
2	Yes	Yes	8000 km	Low+	*3.96*
3	Yes	Yes	8000 km	Medium-	*4.41*
4	Yes	Yes	8000 km	Medium+ & High-	4.62
5	Yes	Yes	8000 km	High+	5.36
6	Yes	Yes	12000 km	Low-	*4.01*
7	Yes	Yes	12000 km	Low+	*4.56*
8	Yes	Yes	12000 km	Medium-	5.07
9	Yes	Yes	12000 km	Medium+ & High-	5.32
10	Yes	Yes	12000 km	High+	6.17
11	Yes	Yes	16000 km	Low-	*4.48*
12	Yes	Yes	16000 km	Low+	5.09
13	Yes	Yes	16000 km	Medium-	5.66
14	Yes	Yes	16000 km	Medium+ & High-	5.94
15	Yes	Yes	16000 km	High+	6.89
16	Yes	Yes	20000 km	Low-	5.34
17	Yes	Yes	20000 km	Low+	6.07
18	Yes	Yes	20000 km	Medium-	6.75
19	Yes	Yes	20000 km	Medium+ & High-	7.08
20	Yes	Yes	20000 km	High+	**8.92**
21	Yes	Yes	30000 km	Low-	6.09
22	Yes	Yes	30000 km	Low+	6.92
23	Yes	Yes	30000 km	Medium-	7.70
24	Yes	Yes	30000 km	Medium+ & High-	**8.07**
25	Yes	Yes	30000 km	High+	**10.18**
26	Yes	Yes	no limit	Low-	6.86
27	Yes	Yes	no limit	Low+	7.79
28	Yes	Yes	no limit	Medium-	**8.67**
29	Yes	Yes	no limit	Medium+ & High-	**9.87**
30	Yes	Yes	no limit	High+	**12.45**

Table 6: Part of the GLM predictions in a decision table for the `norauto` dataset.

References

A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, R. Chatila, and F. Herrera. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, 2020.

J. Ba and R. Caruana. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems 27*, pages 2654–2662, 2014.

P. Biecek. DALEX: explainers for complex predictive models in R. *The Journal of Machine Learning Research*, 19(1):3245–3249, 2018.

C. Bucilă, R. Caruana, and A. Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

D. Doran, S. Schulz, and T. R. Besold. What does explainable AI really mean? A new conceptualization of perspectives. *arXiv preprint arXiv:1710.00794*, 2017.

C. Dutang and A. Charpentier. *CASdatasets: Insurance datasets*, 2019. URL `http://cas.uqam.ca`. R package version 1.0.10.

ECOA. U.S. Code Title 15. Commerce and Trade. *Chapter 41. Consumer Credit Protection*, Subchapter IV. Equal Credit Opportunity (Section 1691), 1974.

J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

J. H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, 2002.

J. H. Friedman and B. E. Popescu. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3):916–954, 2008.

GDPR. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data. *O.J. (L 119)*, 1:1–88, 2016.

B. Greenwell, B. Boehmke, J. Cunningham, and GBM Developers. *gbm: Generalized Boosted Regression Models*, 2019. URL `https://cran.r-project.org/package=gbm`. R package version 2.1.6.

R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. A survey of methods for explaining black box models. *ACM Computing Surveys*, 51(5):1–42, 2018.

D. Gunning. Explainable Artificial Intelligence (XAI). *Defense Advanced Research Projects Agency (DARPA)*, Tech. rep., 2017.

T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference, and prediction.* Springer, New York, 2009.

R. Henckaerts. *maidrr: Model-Agnostic Interpretable Data-driven suRRogate*, 2020. URL `https://github.com/henckr/maidrr`. R package version 1.0.0.

R. Henckaerts, M. P. Côté, K. Antonio, and R. Verbelen. Boosting insights in insurance tariff plans with tree-based machine learning methods. *North American Actuarial Journal*, Accepted, 2020. URL `https://arxiv.org/abs/1904.10890`.

G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

J. Huysmans, K. Dejaeger, C. Mues, J. Vanthienen, and B. Baesens. An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. *Decision Support Systems*, 51(1):141–154, 2011.

J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1:281–297, 1967.

C. Molnar. Interpretable machine learning: A guide for making black box models explainable. 2020. URL `https://christophm.github.io/interpretable-ml-book/`.

C. Molnar, B. Bischl, and G. Casalicchio. iml: An R package for interpretable machine learning. *Journal of Open Source Software*, 3(26):786, 2018.

NAIC. *Model 777 - Guideline 1775 - Guideline 1780 - Product filing review handbook*, 2012. URL `https://naic.org/prod_serv_model_laws.htm`.

J. A. Nelder and R. W. M. Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384, 1972.

C. O'Neil. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown Publishing Group, New York, 2016.

PRIIPs. Regulation (EU) 1286/2014 of the European Parliament and of the Council of 26 November 2014 on key information documents for packaged retail and insurance-based investment products. *O.J. (L 352)*, 1:1–23, 2014.

L. S. Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28): 307–317, 1953.

J. Song. *Ckmeans.1d.dp: Optimal, fast, and reproducible univariate clustering*, 2019. URL `https://cran.r-project.org/package=Ckmeans.1d.dp`. R package version 4.3.0.

E. Štrumbelj and I. Kononenko. An efficient explanation of individual classifications using game theory. *Journal of Machine Learning Research*, 11:1–18, 2010.

E. Štrumbelj and I. Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665, 2014.

H. Wang and M. Song. Ckmeans.1d.dp: optimal K-means clustering in one dimension by dynamic programming. *The R journal*, 3(2):29, 2011.