When stakes are high: balancing accuracy and transparency with Model-Agnostic Interpretable Data-driven suRRogates

Roel Henckaerts, Katrien Antonio, Marie-Pier Côté

When stakes are high: balancing accuracy and transparency with Model-Agnostic Interpretable Data-driven suRRogates

Roel Henckaerts [*,b,d], Katrien Antonio[b,c,d], and Marie-Pier Côté[a]

[a] *École d'actuariat, Université Laval, Canada.*
[b] *Faculty of Economics and Business, KU Leuven, Belgium.*
[c] *Faculty of Economics and Business, University of Amsterdam, The Netherlands.*
[d] *LRisk, Leuven Research Center on Insurance and Financial Risk Analysis, KU Leuven, Belgium.*

Abstract

Highly regulated industries, like banking and insurance, ask for transparent decision-making algorithms. At the same time, competitive markets are pushing for the use of complex black box models. We therefore present a procedure to develop a Model-Agnostic Interpretable Data-driven suRRogate (maidrr) suited for structured tabular data. Knowledge is extracted from a black box via partial dependence effects. These are used to perform smart feature engineering by grouping variable values. This results in a segmentation of the feature space with automatic variable 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 publicly available datasets. Our maidrr GLM closely approximates a gradient boosting machine (GBM) black box and outperforms both a linear and tree surrogate as benchmarks.

Key words: Compliance, Feature selection, GLM, Insurance, Segmentation, XAI

1 Introduction

The big data revolution opened the door to highly complex artificial intelligence (AI) technology in search for top performance. However, at the same time, there is growing public awareness for the issues of interpretability, explainability and fairness of AI systems (O'Neil, 2016). The General Data Protection Regulation (GDPR, 2016) introduces "the right to an explanation" of decision-making algorithms, thereby pushing for transparent communication on the underlying rationale of the decisions. An explainable AI (XAI) algorithm enables human users to understand, trust and manage its decisions (Gunning, 2017). Explainability is gaining attention in many industries, such as automotive (Meteier et al., 2019), banking (Bracke et al., 2019), healthcare (Ahmad et al., 2018), insurance (OECD, 2020), manufacturing (Hrnjica and Softic, 2020) and critical systems (Gade et al., 2019). Full transparency is essential for high-stakes decisions with a big impact on a person's life, such as medical diagnosis, insurance coverage, education admission, loan applications, criminal justice, autonomous transportation and job recruitment.

A lack of algorithmic transparency can hinder AI implementations in business practice due to regulatory compliance requirements (Arrieta et al., 2020). XAI is therefore especially important in highly regulated industries with an extensive review of algorithms by supervisory authorities. Examples from the financial sector include 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). Our case study in Section 3 puts focus on general insurance pricing as one of the high-stakes XAI application areas where transparent decision-making is essential due to strict regulations.

*Corresponding author: roel.henckaerts@kuleuven.be.

A clear distinction regarding model explainability is made between interpretation techniques *ex-post* and transparency *ex-ante* (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 transparent by design, meaning they are easily comprehensible for human users. In linear models, 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).

Surrogate models intend to copy the behavior of a complex system by capturing its essence in a simpler format. This 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 approaches 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. The underlying structure of the complex system is learned by using its predictions as labels for training the surrogate. Within XAI applications, an interpretable surrogate is used to explain the complex system. Global surrogates explain average model behavior for a given dataset (Molnar et al., 2020). Local surrogates, such as LIME (Ribeiro et al., 2016), K-LIME (Hall et al., 2017), SHAP (Lundberg and Lee, 2017), Anchors (Ribeiro et al., 2018) and SLIM (Hu et al., 2020), explain individual predictions by an interpretable model in the vicinity of the observation of interest.

This paper presents a procedure to develop a global surrogate for a complex system, with the goal of implementing the surrogate in production. The surrogate inherits the strengths of a sophisticated black box algorithm, delivered in a simpler format that is easier to understand, manage and implement. The resulting high degree of model transparency can boost AI business applications, especially in highly regulated sectors such as banking and insurance. Our procedure extracts knowledge from the complex system via ex-post interpretation techniques. Next, using these insights, it performs smart feature engineering on the training data. In the end, an ex-ante transparent surrogate is fit to the engineered training data. The surrogate closely approximates the black box model such that it can be used as a substitute with explanations readily available.

We put forward the following three desirable properties. 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 knowledge from the black box, thereby covering a vast amount of different model types (Friedman, 2001). Secondly, the resulting surrogate should be *interpretable*, making it easy to comprehend and use by human users. We employ generalized linear models (GLMs), formulated by Nelder and Wedderburn (1972). This versatile model class covers a broad range of classification and regression models and allows to represent its output as a decision table. GLMs are therefore widely used in for example the insurance industry. Thirdly, a *data-driven* procedure avoids the need for ad hoc model choices. We fully automate the transformation from black box to transparent surrogate via a cross-validation scheme.

We introduce maidrr: a Model-Agnostic Interpretable Data-driven suRRogate procedure for a black box developed on structured tabular data. The complete procedure is available in the open source R package `maidrr` (Henckaerts, 2020). 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, where transparency is essential due to strict regulations. We demonstrate that our maidrr surrogate GLM is able to approximate the performance of a black box closely, while outperforming 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 transform into a simpler and more comprehensible surrogate. We extract knowledge 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 may 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. After this step of feature engineering, a generalized linear model (GLM) is fit to the segmented data with all features in a categorical format and their relevant interactions. The end product is an interpretable surrogate which approximates the black box model.

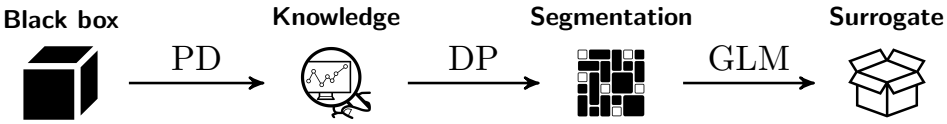


Figure 2: The maidrr process for transforming a black box algorithm into a transparent GLM.

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.

Knowledge A univariate partial dependence (PD) captures the marginal relation between a feature x_j, for $j \in \{1, \ldots, p\}$, and the model predictions (Friedman, 2001). The PD effect $\bar{f}_j(x_j)$ evaluates the prediction function f_{pred} for a given value of feature x_j, while averaging over n observed values of the other features $\boldsymbol{x}^i_{-j}$ for observation $i \in \{1, \ldots, n\}$:

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

The PD effect $\bar{f}_j$ is used to group values/levels within feature x_j, as a similar PD indicates a similar relation to the prediction target. This grouping reduces 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 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 $z_{j,q}$. This represents a one-dimensional clustering problem of $z_{j,q}$ for $q \in \{1, \ldots, m_j\}$. In theory, PDs can be misleading for correlated features and accumulated local effects (ALE) serve as an alternative (Apley and Zhu, 2019). However, Appendix A compares the resulting PDs and ALEs for highly correlated features, justifying the use of PDs for grouping purposes.

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. We impose such constraints for continuous/ordinal features in order 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 can perform a p-dimensional grid search to find the optimal k_j 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. We propose a penalized loss function to find the optimal number of groups k_j.

Penalized loss function 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 set of values, as follows:

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

The first part of this loss function measures how well the PD effect is approximated by the grouped variant as a weighted mean squared error (wMSE) over all unique values of feature x_j. The weight $w_{j,q}$ represents the proportion of observations that equal value $x_{j,q}$ for feature x_j. This forces the procedure to focus on closely approximating frequently occurring feature values as opposed to rare cases. 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. The optimal λ value is determined via cross-validation, as detailed in Paragraph Hyperparameters.

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 knowledge, 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. We refer to Appendix B for details on the GLM formulation. GLMs with only categorical features lead to fixed-size decision tables, see Appendix C for an example. Even with many features they remain transparent, fileable in a tabular format and easy to use by business intermediaries, so the complexity of the GLM is not a concern. The high degree of transparency, thanks to observable coefficients, allows intuitive model post-processing by industry experts when necessary. GLMs are therefore attractive when transparency is essential, they are for example the preferred pricing tool within the strictly regulated insurance industry.

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

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

We define feature $x_{a:b}$ as the interaction containing all combinations of features x_a and x_b in the original data. The DP algorithm clusters levels in $x_{a:b}$ that have similar $\bar{f}_{a,b}(x_a, x_b)$ values, without any adjacency constraints. Interactions represent a correction on top of the marginal effects so we 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). The grouped version of $x_{a:b}$ enters the surrogate GLM in a categorical format.

Hyperparameters Algorithm 1 details the full maidrr procedure with four input parameters: λ_{marg}, λ_{intr}, k and h. A distinct value of λ is advised for marginal and interaction effects, as the PDs in Eq. (1) and (3) reside on different scales. Marginal PDs are expressed on the scale of the predictions, whereas interaction PDs are expressed as a pure interaction effect. 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 a loss function for the surrogate GLM predictions. This loss 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. We refer to Section 3.2.2 for more details. Automatic feature selection is enabled as feature x_j is excluded from the surrogate when $k_j = 1$. The hyperparameter k allows to specify a maximum number of groups for feature segmentation. The hyperparameter h selects a set of relevant interactions by means of a cut-off on the realized values of the black box's H-statistic, thereby excluding unimportant interactions upfront.

Algorithm 1 maidrr

Input: data, f_{pred}, λ_{marg}, λ_{intr}, k and h

marginal:
- **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 $k_j^* = \underset{k_j\in\{1,\ldots,k\}}{\arg\min}$ Eq. (2) for $\lambda = \lambda_{\text{marg}}$
  - x_j^c represents the grouped version of x_j in categorical format with k_j^* groups
- **end for**
- feature selection: $F = \{j \mid k_j^* > 1\}$

interaction:
- upfront interaction selection: $I = \{(l, m) \mid l \in F$ **and** $m \in F$ **and** $H(x_l, x_m) \geq h\}$
- **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 $k_{ab}^* = \underset{k_j\in\{1,\ldots,k\}}{\arg\min}$ Eq. (2) for $\lambda = \lambda_{\text{intr}}$
  - $x_{a:b}^c$ represents the grouped version of $x_{a:b}$ in categorical format with k_{ab}^* groups
- **end for**
- interaction selection: $I = I \setminus \{(l, m) \mid k_{lm}^* = 1\}$

fit a GLM to the target with features x_j^c for $j \in F$ and interactions $x_{a:b}^c$ for $(a, b) \in I$

Output: surrogate GLM

3 Case study for the insurance industry

In most jurisdictions, insurers are required by law to document their pricing or rating model to the regulator. Determining a fair insurance quote is also high-stakes, as it can have a big impact on a person's life. This creates a clear need for transparency in the underlying decision-making process. 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 maidrr GLM surrogate. Section 3.3 evaluates the 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 `CASdatasets` (Dutang and Charpentier, 2019) or `maidrr` (Henckaerts, 2020). 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).

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

	`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

*The name of the dataset corresponds to its name in the R package.

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, the 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 are proportional to exposure and have a multiplicative structure: $\mathbb{E}[y] = t\times\exp(\beta_0)\times\prod_j \exp(\beta_j x_j)$.

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

	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

3.2 Finding a transparent model by opening the black box

Section 3.2.1 describes the construction of a gradient boosting machine or GBM as black box. Section 3.2.2 details the maidrr procedure to obtain a GLM surrogate and illustrates the automatic feature selection and segmentation for several datasets.

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 good performance of GBMs as discussed 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 follows:

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

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

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

3.2.2 GLM surrogate via maidrr

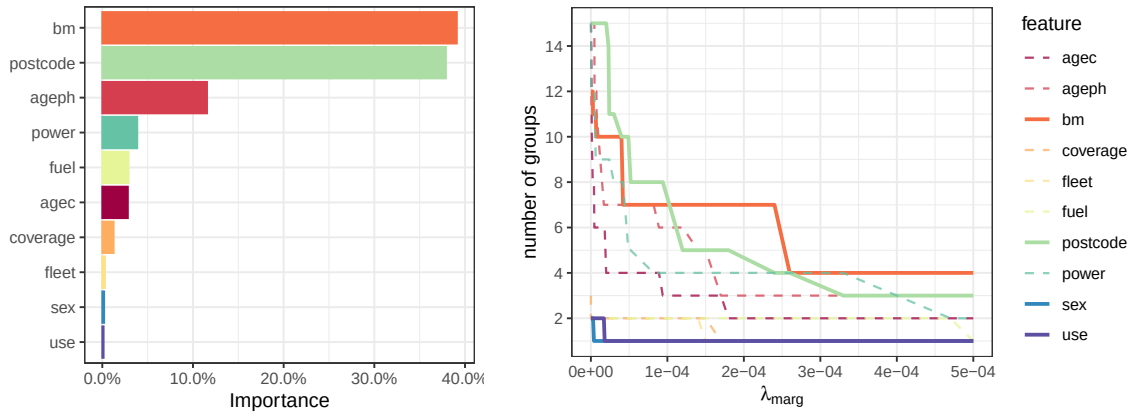
We build a surrogate GLM to approximate the optimal GBM for each dataset. The function `maidrr::autotune` (Henckaerts, 2020) implements a tuning procedure for Algorithm 1.

Algorithm 1 requires four input parameters: λ_{marg}, λ_{intr}, k 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^{-10} to 1. This range is sufficiently wide for our application, as indicated by the optimal values in Table 4. Tuning of the λ values is done in two stages. First, a grid search over λ_{marg} finds the optimal GLM with only marginal effects by running the "marginal" part of Algorithm 1. Then, a grid search over λ_{intr} determines which interactions to include in that optimal GLM by running the "interaction" part of Algorithm 1. This requires two one-dimensional grid searches of length $grid\_size$ instead of one two-dimensional search of length $grid\_size^2$, thereby saving computation time. The optimal λ values are determined by performing 5-fold cross-validation on the resulting GLM with the Poisson deviance in Eq. (4) as loss function. 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. We set the maximum number of groups $k = 15$.

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

	`ausprivauto`	`bemtpl`	`freMPL`	`freMTPL`	`norauto`	`pricingame`
λ_{marg}	4.2×10^{-5}	4.2×10^{-5}	1.6×10^{-4}	1.3×10^{-7}	1.1×10^{-5}	2.0×10^{-6}
λ_{intr}	8.5×10^{-6}	4.1×10^{-6}	4.6×10^{-5}	3.1×10^{-6}	3.1×10^{-5}	2.8×10^{-6}

Figure 3 illustrates the automatic feature selection of maidrr for the `bemtpl` portfolio. Figure 3(a) shows feature importance scores according to the GBM and Figure 3(b) shows the number of groups for each feature in function of λ_{marg}. Important features, such as `bm` and `postcode`, retain a higher number of groups for increasing values of λ_{marg}. Levels of uninformative features, like `use` and `sex`, are quickly placed in one group, effectively excluding these variables from the GLM. This is how maidrr performs automatic feature selection via the data-driven tuning of λ_{marg}.



(a) Feature importance in the optimal GBM

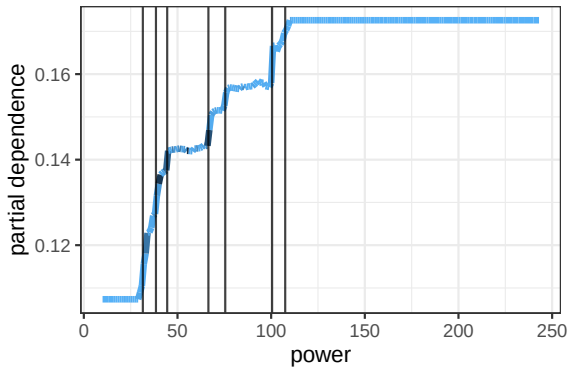
(b) Number of groups in function of λ_{marg}

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

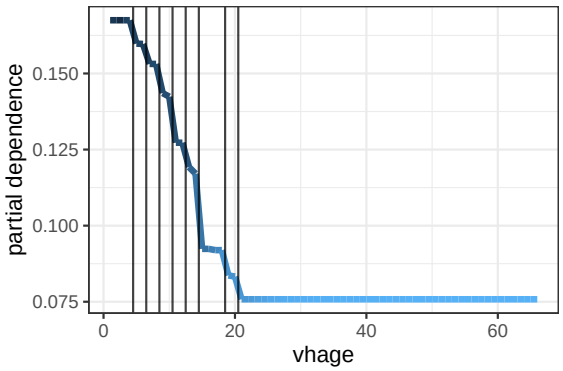
Figure 4 displays the resulting segmentation for two continuous features: vehicle power for `bemtpl` in Figure 4(a) and vehicle age for `pricingame` in Figure 4(b). Both show the GBM PD effect, where darker blue indicates a higher observation count in the portfolio. The features are grouped into 8 and 9 bins respectively, indicated by the vertical lines. The bins are wide wherever the PD effect is quite stable and narrow where the effect is steeper. We observe that claim risk increases for increasing vehicle power, while it decreases for increasing vehicle age.

Figure 5 displays the resulting segmentation for three categorical features. Groups are indicated by different plotting characters, with size proportional to the observation count in the portfolio. Figure 5(a) shows that claim risk decreases with increasing age of the policyholder in the `ausprivauto` portfolio. Due to similar PD effects, both levels containing the oldest policyholders are grouped together as well as both levels containing the people of working age. This results in four age segments: youngest, young, working and older people. Figure 5(b) shows that claim risk decreases for a decreasing driving distance limit in the `norauto` portfolio. The PD effects are dissimilar enough not to be grouped together, so each level remains in a separate segment. Figure 5(c) shows the PD effects and resulting grouping for vehicle makes in the `norauto` portfolio. The 41 different makes are divided in 11 segments with {Mazda, Jeep} and {Lada, Unic, Other} as the most and least risky segments respectively. Categorical features with many levels are often hard to deal with in practice. Appendix D demonstrates how maidrr greatly reduces

the complexity for geographical information in the `bemtpl` and `pricingame` portfolios.

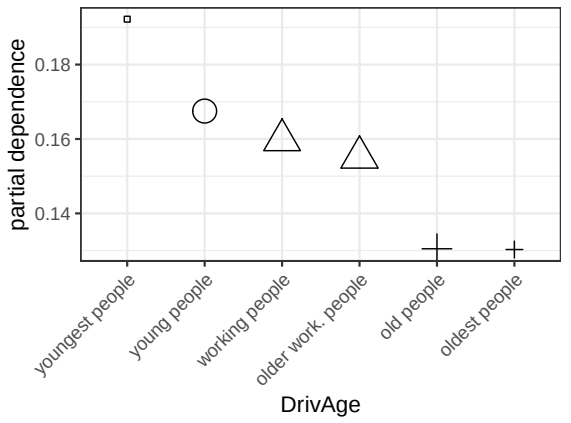


(a) `bemtpl`: power of the vehicle in hp

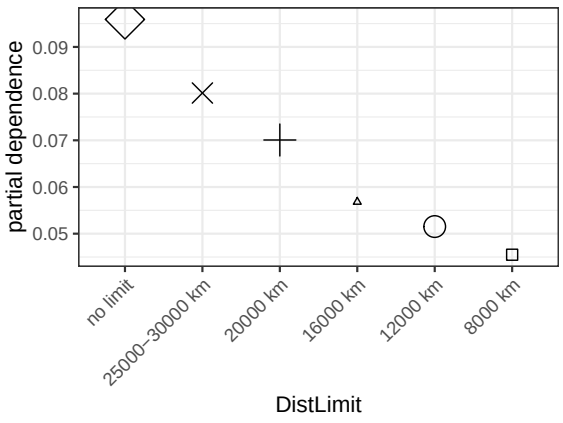


(b) `pricingame`: age of the vehicle in years

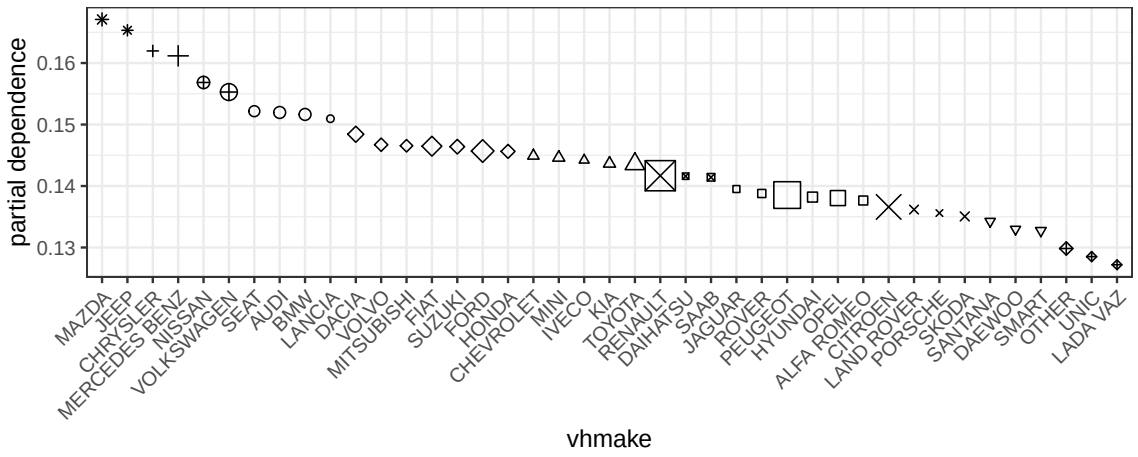
Figure 4: PD effect and the resulting segmentation for two continuous features. Groups are separated by vertical lines and darker blue indicates a higher observation count in the portfolio.



(a) `ausprivauto`: age of the policyholder



(b) `norauto`: driving distance limit



(c) `pricingame`: make of the vehicle

Figure 5: PD effect and the resulting segmentation for three categorical features. Groups are indicated by plotting characters, with size proportional to the observation count in the portfolio.

3.3 Evaluation of the GLM surrogate

This section evaluates the performance of the maidrr GLM surrogate based on three desiderata listed in Guidotti et al. (2018, Section 3.2): accuracy, fidelity and interpretability. Section 3.3.1 evaluates accuracy since generating accurate predictions is very important for a model to remain competitive and relevant in production. Section 3.3.2 evaluates fidelity as the extent to which the surrogate is able to mimic the behavior of a black box. Section 3.3.3 evaluates interpretability because the surrogate should be comprehensible and easy to use in practice. 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

The goal of our maidrr GLM surrogate is to approximate a complex black box and replace it in the production pipeline. In order to justify this substitution, it is vital that the GLM results in accurate predictions with minimal accuracy loss compared to the black box. 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 percentage differences as follows: $\Delta D^{\text{Poi}} = 100 \times \left(D^{\text{Poi}}\{y, f_{\text{surro}}(\boldsymbol{x})\}/D^{\text{Poi}}\{y, f_{\text{gbm}}(\boldsymbol{x})\} - 1\right)$.

Table 5 shows the Poisson percentage differences ΔD^{Poi} for the GLM, LM and DT surrogates with respect to the GBM black box. Results are shown for each dataset separately and the last column contains the average over all datasets. The maidrr GLM attains the lowest accuracy loss and outperforms the benchmark surrogates on each dataset. The GLM's accuracy loss stays below 0.5% on four out of six datasets, with an average of 0.64% over all datasets. On average, the GLM is 3 and 7.5 times as accurate as the DT and LM surrogates.

Table 5: Poisson percentage differences ΔD^{Poi} for the different surrogate models.

	`ausprivauto`	`bemtpl`	`frempl`	`fremtpl`	`norauto`	`pricingame`	avg.
GLM	**0.10**	**0.49**	**1.80**	**0.92**	**0.03**	**0.48**	**0.64**
LM	0.22	1.15	18.39	6.35	0.07	2.53	4.79
DT	0.25	1.68	4.82	2.66	0.28	2.13	1.97

3.3.2 Fidelity

This section investigates how closely the maidrr GLM mimics the behavior of the GBM black box by assessing how well the surrogates replicate the GBM's predictions.

The R^2 measure represents the percentage of variance that the surrogate model is able to capture from the black box. With μ_{gbm} the mean GBM prediction, the $R^2 \in [0, 1]$ is defined as follows:

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

Furthermore, we also compute Pearson's linear and Spearman's rank correlation coefficients ρ between the GBM and surrogate predictions. We average these coefficients to consolidate both types of correlation in one number, but the results below also hold for each coefficient separately.

Table 6 shows the R^2 for the GLM, LM and DT surrogates on each dataset separately and averaged over all datasets in the last column. The GLM ranks first in five datasets and second in `ausprivauto`. The GLM captures more than 90% of variance on four out of six datasets, with an average of 90% over all datasets. On average, the GLM captures an extra 12% and 15% of variance compared to the DT and LM surrogates.

Table 6: R^2 measure for the different surrogate models.

	`ausprivauto`	`bemtpl`	`frempl`	`fremtpl`	`norauto`	`pricingame`	avg.
GLM	0.86	**0.94**	**0.91**	**0.78**	**0.99**	**0.93**	**0.90**
LM	**0.89**	0.83	0.62	0.30	0.95	0.88	0.75
DT	0.75	0.74	0.88	0.75	0.84	0.76	0.78

Table 7 shows the averaged ρ for the GLM, LM and DT surrogates on each dataset separately and averaged over all datasets in the last column. The GLM ranks first in all datasets, thereby outperforming both benchmark surrogates. The correlation between the GBM and GLM is at least 95% on four out of six datasets, with an average of 95% over all datasets. On average, the GLM's correlation to the GBM is 12% and 9% higher compared to the DT and LM surrogates.

Table 7: Average correlation coefficient ρ for the different surrogate models.

	`ausprivauto`	`bemtpl`	`frempl`	`fremtpl`	`norauto`	`pricingame`	avg.
GLM	**0.95**	**0.97**	**0.91**	**0.92**	**0.99**	**0.97**	**0.95**
LM	0.95	0.93	0.74	0.60	0.98	0.95	0.86
DT	0.86	0.83	0.75	0.78	0.91	0.87	0.83

In both Tables 6 and 7, the DT outperforms the LM on the `frempl` and `fremtpl` datasets while the LM outperforms the DT on the remaining four datasets. This is driven by the fact that the DT puts focus on interactions while the LM puts focus on marginal effects. Our maidrr GLM combines both marginal and interaction effects, resulting in better performance overall.

We conclude that our GLM constructed with maidrr outperforms the benchmark DT and LM surrogates when it comes to both prediction accuracy and mimicking the GBM's underlying behavior. Remember that the DT and LM are trained with the GBM's predictions as target. The maidrr procedure extracts knowledge from the GBM to perform smart feature engineering, but afterwards the GLM is fit to the original target. The observation that the GLM is better at mimicking the GBM compared to the benchmark surrogates is therefore especially interesting.

3.3.3 Interpretability

Global interpretations A GLM is globally interpretable as the model coefficients, relating the features to the predictions, are easily observable. For example, the Poisson GLM with logarithmic link function to model the number of claims for the `norauto` dataset has the following structure and fitted coefficients:

$$\begin{aligned}\ln\left(\mathbb{E}[nclaims]\right) = &-2.40 + 0.54\,Male_{no} + 0.09\,Young_{yes} \\ &- 0.76\,DistLimit_{8000km} - 0.62\,DistLimit_{12000km} \\ &- 0.51\,DistLimit_{16000km} - 0.33\,DistLimit_{20000km} - 0.20\,DistLimit_{30000km} \\ &- 0.17\,GeoRegion_{Low- \;\&\; Low+} - 0.05\,GeoRegion_{Med-} + 0.23\,GeoRegion_{High+} \\ &- 0.08\,DistLimit\_\_GeoRegion_{8000/12000/16000km\_\_High+ \;\&\; nolimit\_\_Low-/Low+/Med-}\end{aligned}$$

where $Male_{yes}$, $Young_{no}$, $DistLimit_{nolimit}$ and $GeoRegion_{Med+\ \&\ High-}$ are the reference levels captured by the intercept. These references are the levels which contain the highest number of policyholders such that the intercept models the claim frequency of an "average" policyholder. Taking the inverse link function, namely the exponential, on both sides results in a multiplicative GLM prediction function with the following global interpretations:

- The predicted claim frequency for an older male policyholder without a driving distance limit and living in the Med+ or High- geographical region equals 0.09 or $\exp(-2.40)$.
- Predictions are 72% higher for female policyholders compared to males as $\exp(0.54) = 1.72$. *Note: In 2012, the EU put forward rules on gender-neutral pricing in the insurance industry such that gender is no longer allowed as a rating factor in a commercial tariff.*
- As $\exp(0.09) = 1.09$, predictions are 9% higher for young compared to old policyholders.
- For policyholders with a driving distance limit of 8, 12, 16, 20 and 30 thousand kilometers, predictions respectively amount to 47%, 54%, 60%, 72% and 82% of those for someone without a limit. There is a clear increasing trend of claim risk with the distance limit.
- Predictions for policyholders living in the Low or Med- geographical regions amount to respectively 84% and 95% of those for the Med+/High- regions, whereas predictions increase with 26% for those in the High+ region.
- The interaction between the distance limit and geographical region results in a negative correction for policyholders with the most risky level of one of the features and a low-risk level of the other. As $\exp(0.08) = 0.92$, predictions are reduced by 8% for policyholders living in the High+ region with a maximal distance limit of 16,000 kilometer and for those with no distance limit which live in the Low-, Low+ or Med- region.

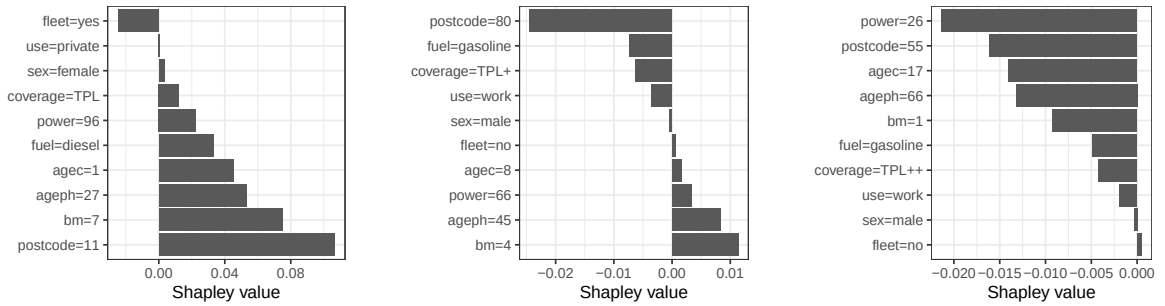
Our maidrr procedure outputs a GLM with all features in a categorical format such that the full working regime of the GLM can be summarized in a decision table, see Appendix C. In practice, such a tabular model is easy to represent and maintain in a spreadsheet with responsive filters. Decision tables are very comprehensible for human users and outperform both trees and rules in accuracy, response time, answer confidence and ease of use (Huysmans et al., 2011).

Local interpretations We now turn to explaining individual predictions for the three artificial instances in the `bemtpl` dataset listed in Table 8. Based on the GBM and GLM predictions, these instances represent a high/medium/low risk profile. We want to assess how the features influence the riskiness of each individual. Feature contributions in a GLM can be extracted via the fitted coefficients, as implemented in `maidrr::explain` (Henckaerts, 2020). For comparison purposes we use Shapley (1953) values to explain the GBM predictions, with the efficient implementation of Štrumbelj and Kononenko (2010, 2014) available in the R package `iml` (Molnar et al., 2018).

Figures 6(a), 6(b) and 6(c) show the Shapley values for the GBM prediction of each instance. The sum of these values equals the difference between the instance prediction, shown in Table 8, and the average GBM prediction of 0.1417. The presence of mainly positive/negative Shapley values in Figure 6(a)/6(c) thus represents a high/low risk profile respectively. Figures 6(d), 6(e) and 6(f) show the GLM's feature contributions on the response scale after taking the inverse link function, namely $\exp(\beta_j)$ for feature x_j in our Poisson GLMs with log link. The contributions are multiplicative with respect to the baseline prediction of 0.13, as captured by the intercept, and the gray dashed line indicates the point of "no contribution" at $\exp(0)$. Furthermore, the

Table 8: Artificial instances in the `bemtpl` portfolio for which we explain the individual predictions.

	high risk	medium risk	low risk
`bm`	7	4	1
`postcode`	11	91	55
`ageph`	27	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`	private	work	work
GBM	0.2847	0.1398	0.0502
GLM	0.3861	0.1231	0.0413



(a) GBM: high risk (b) GBM: medium risk (c) GBM: low risk

(d) GLM: high risk (e) GLM: medium risk (f) GLM: low risk

Figure 6: Explanations for the high (left), medium (middle) and low (right) risk instance predictions from Table 8 in the GBM via Shapley values (top) and the GLM via β coefficients (bottom).

GLM allows to split the contributions over marginal effects and interactions with other features, while 95% confidence intervals indicate the uncertainty associated with each contribution.

The GBM and GLM explanations are very similar. For example, Figures 6(a) and 6(d) attribute this profile's high risk to a residence in Brussels, young age, high bonus-malus level and driving a new high-powered diesel vehicle. The interaction between the bonus-malus level and age of the vehicle puts a negative correction on both positive marginal effects in the GLM, while the other interactions have limited impact on the prediction. The GLMs show no contribution from gender as this feature is not selected by maidrr, while it has negligibly small Shapley values in all cases. An insurance rate is determined by the product of claim frequency and severity, such that the contributions can be directly interpreted as a percentage premium/discount on the price. Living in Brussels increases the baseline frequency, and thus the price, by almost 50% in the

technical analysis for 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

Decision-making algorithms in business practice can become highly complex in order to gain a competitive advantage. However, transparency is a key requirement for any high-stakes decision or for companies active in strictly regulated industries. To balance accuracy and explainability, we present maidrr: a procedure to develop a Model-Agnostic Interpretable Data-driven suRRogate for a complex system. The paper is accompanied by an R package in which the procedure is implemented (Henckaerts, 2020). We apply maidrr to six real-life general insurance portfolios for claim frequency prediction, with insurance pricing as an example of a high-stakes decision in a strictly regulated industry. We thereby also put focus on a highly relevant count regression problem, which is not often dealt with in classical machine learning. Our maidrr procedure results in a surrogate GLM which closely approximates the performance of a black box GBM in terms of accuracy and fidelity, while outperforming two benchmark surrogates. This allows to substitute the GLM in the production pipeline with minimal performance loss. In the process, maidrr automatically performs feature selection and segmentation, providing a possibly useful by-product for customer or market segmentation applications.

Both global and local interpretations are easily extracted from our maidrr GLM. Explanations only depend on the fitted coefficients, which are easily observable and presentable on the response scale. 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. 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 for high-stakes decisions which influence 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 serves 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.

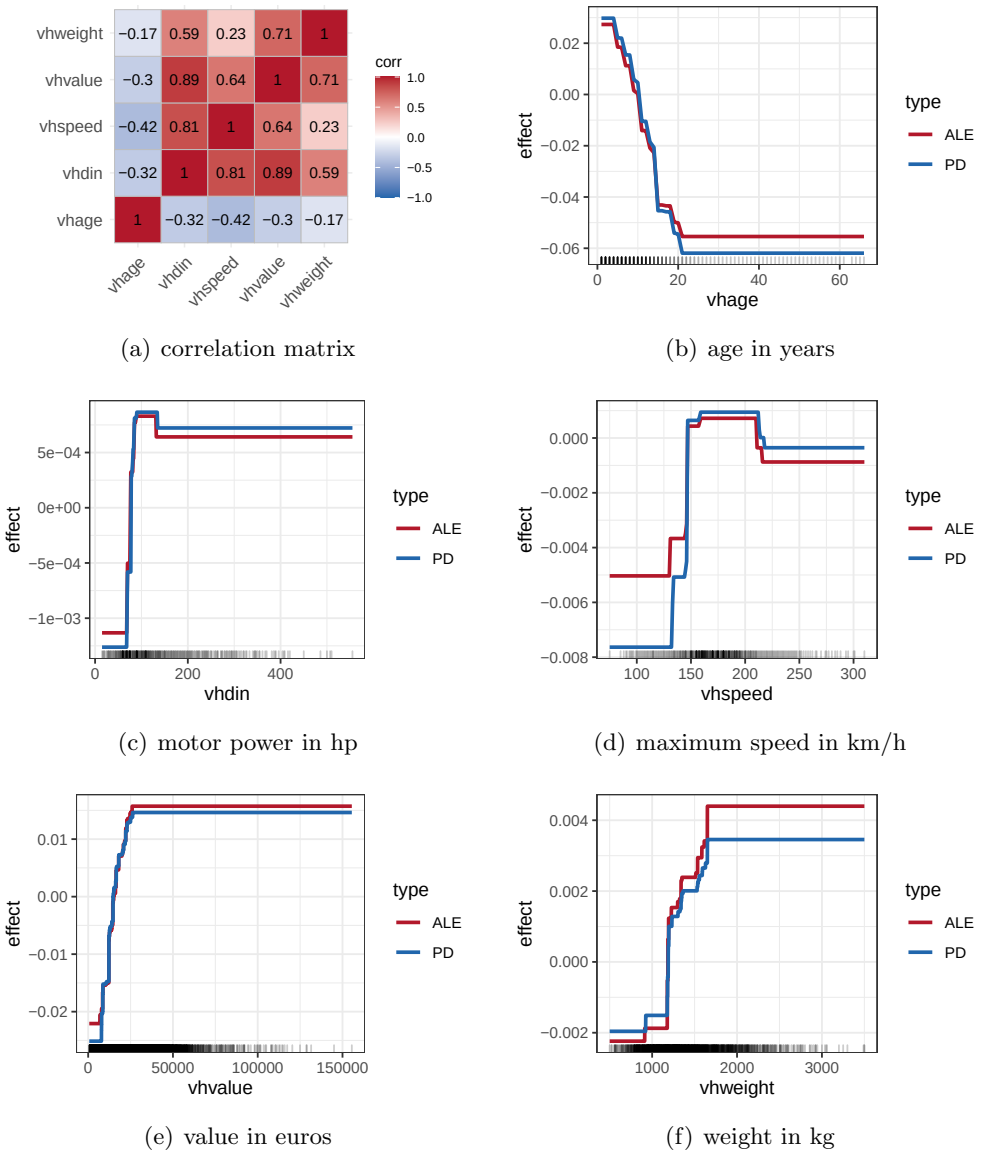
Our maidrr procedure combines the inherent interpretability of a GLM with the accurate approximation of a sophisticated black box. We therefore believe that maidrr can serve as a useful tool in any situation where a competitive, yet transparent model is needed.

Funding

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].

A PD and ALE for correlated features

Figure 7 compares the PD and ALE for several vehicle characteristics in the `pricingame` dataset, namely the weight, value, maximum speed, horsepower and age. Figure 7(a) shows that the vehicle age is negatively correlated with the other characteristics while there is a strong positive correlation between the weight, value, maximum speed and horsepower. Figures 7(b), 7(c), 7(d), 7(e) and 7(f) show the centered PD (in blue) and ALE (in red) for all the vehicle features. Both effects are very similar for each of the features, especially in the ranges with high observation counts as indicated by the black rugs on the x-axis. We observe some vertical shifts between the PD and ALE in feature ranges with low observation counts. However, these vertical shifts are not a problem for our maidrr procedure as we only use these effects to perform the feature grouping. Furthermore, observation counts are taken into account as weights in the penalized loss function of Eq. (2), further reducing the impact of these shifts on the obtained segmentation. This justifies the use of PD effects for grouping, even when dealing with correlated features.



(a) correlation matrix

(b) age in years

(c) motor power in hp

(d) maximum speed in km/h

(e) value in euros

(f) weight in kg

Figure 7: Comparison of PD and ALE for correlated vehicle characteristics in the `pricingame` dataset.

B 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 distributions, making GLMs a versatile modeling tool. Denoting by $g(\cdot)$ the link function, the structure of a GLM with all features $\boldsymbol{x}$ in a categorical format is as follows:

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

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.

C GLM in a tabular format

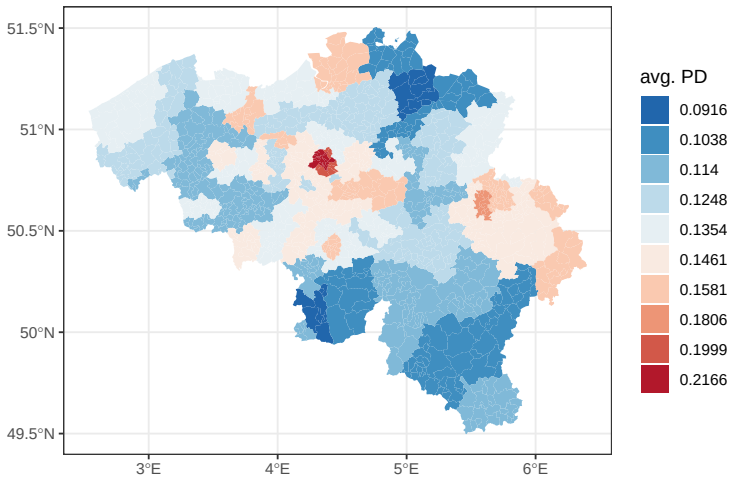
Table 9 shows part of the decision table for the `norauto` dataset, with the four lowest and highest predictions indicated in italics and bold respectively. The three other parts for *Male = Yes & Young = No* and *Male = No & Young = Yes/No* are not shown for space reasons.

Table 9: Part of the GLM predictions in a decision table for the `norauto` dataset.

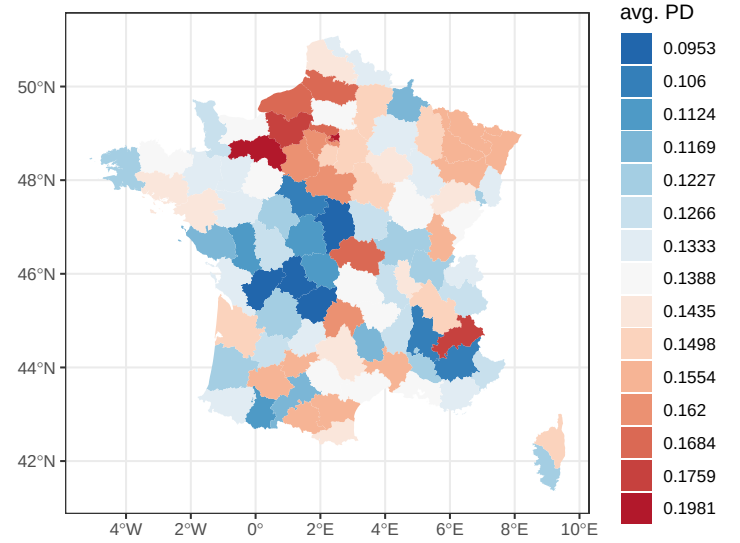
	Male	Young	DistLimit	GeoRegion	GLM prediction (%)
1	Yes	Yes	8000 km	Low- & Low+	*3.88*
2	Yes	Yes	8000 km	Medium-	*4.41*
3	Yes	Yes	8000 km	Medium+ & High-	*4.62*
4	Yes	Yes	8000 km	High+	5.36
5	Yes	Yes	12000 km	Low- & Low+	*4.47*
6	Yes	Yes	12000 km	Medium-	5.08
7	Yes	Yes	12000 km	Medium+ & High-	5.32
8	Yes	Yes	12000 km	High+	6.17
9	Yes	Yes	16000 km	Low- & Low+	4.99
10	Yes	Yes	16000 km	Medium-	5.67
11	Yes	Yes	16000 km	Medium+ & High-	5.94
12	Yes	Yes	16000 km	High+	6.89
13	Yes	Yes	20000 km	Low- & Low+	5.94
14	Yes	Yes	20000 km	Medium-	6.75
15	Yes	Yes	20000 km	Medium+ & High-	7.07
16	Yes	Yes	20000 km	High+	**8.92**
17	Yes	Yes	30000 km	Low- & Low+	6.78
18	Yes	Yes	30000 km	Medium-	7.70
19	Yes	Yes	30000 km	Medium+ & High-	8.07
20	Yes	Yes	30000 km	High+	**10.18**
21	Yes	Yes	no limit	Low- & Low+	7.63
22	Yes	Yes	no limit	Medium-	8.67
23	Yes	Yes	no limit	Medium+ & High-	**9.87**
24	Yes	Yes	no limit	High+	**12.45**

D Geographical segmentation

Figure 8 shows the average PD effect for geographical regions where groups are indicated by colors. Figure 8(a) shows the postal code areas on the map of Belgium with the initial 80 regions from the `bemtpl` portfolio segmented in 10 clusters. 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. Figure 8(b) shows the INSEE department code areas on the map of France with the initial 96 regions from the `pricingame` portfolio segmented in 15 clusters. The capital Paris and surrounding departments in the north of France (red/orange colored) are high-risk areas.



(a) `bemtpl`: postal code



(b) `pricingame`: INSEE department code

Figure 8: Average PD effect for geographical regions where groups are indicated by colors.

References

M. A. Ahmad, C. Eckert, and A. Teredesai. Interpretable machine learning in healthcare. In *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics*, pages 559–560, 2018.

D. W. Apley and J. Zhu. Visualizing the effects of predictor variables in black box supervised learning models. *arXiv preprint arXiv:1612.08468*, 2019.

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.

P. Bracke, A. Datta, C. Jung, and S. Sen. *Machine learning explainability in finance: an application to default risk analysis*. Bank of England Working Paper No. 816, 2019.

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.

K. Gade, S. C. Geyik, K. Kenthapadi, V. Mithal, and A. Taly. Explainable AI in industry. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3203–3204, 2019.

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.

P. Hall, N. Gill, M. Kurka, and W. Phan. *Machine learning interpretability with H2O Driverless AI*. H2O.ai, 2017. URL http://docs.h2o.ai/driverless-ai/latest-stable/docs/booklets/MLIBooklet.pdf.

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*, pages 1–31, 2020.

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

arXiv:1503.02531, 2015.

B. Hrnjica and S. Softic. Explainable AI in manufacturing: A predictive maintenance case study. In *Advances in Production Management Systems. Towards Smart and Digital Manufacturing*, pages 66–73. Springer, 2020.

L. Hu, J. Chen, V. N. Nair, and A. Sudjianto. Surrogate locally-interpretable models with supervised machine learning algorithms. *arXiv preprint arXiv:2007.14528*, 2020.

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.

S. M. Lundberg and S. Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30, pages 4765–4774, 2017.

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, pages 281–297, 1967.

Q. Meteier, M. Capallera, L. Angelini, E. Mugellini, O. A. Khaled, S. Carrino, E. De Salis, S. Galland, and S. Boll. Workshop on explainable ai in automated driving: a user-centered interaction approach. In *Proceedings of the 11th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pages 32–37, 2019.

C. Molnar. *Interpretable machine learning: A guide for making black box models explainable*. Leanpub, 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.

C. Molnar, G. Casalicchio, and B. Bischl. Interpretable machine learning–a brief history, state-of-the-art and challenges. *arXiv preprint arXiv:2010.09337*, 2020.

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.

OECD. *The Impact of Big Data and Artificial Intelligence (AI) in the Insurance Sector*, 2020. URL https://oecd.org/finance/Impact-Big-Data-AI-in-the-Insurance-Sector.htm.

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.

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

M. T. Ribeiro, S. Singh, and C. Guestrin. Anchors: High-precision model-agnostic explanations. In *AAAI Conference on Artificial Intelligence*, volume 18, pages 1527–1535. AAAI, 2018.

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.