Weight-of-evidence through shrinkage and spline binning for interpretable nonlinear classification

Applied Soft Computing 115 (2022) 108160

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Weight-of-evidence through shrinkage and spline binning for interpretable nonlinear classification

Jakob Raymaekers [a,b], Wouter Verbeke [d], Tim Verdonck [b,c,*]

[a] Department of Quantitative Economics, Maastricht University, Maastricht, The Netherlands
[b] Department of Mathematics, University of Antwerp, Antwerp, Belgium
[c] Department of Mathematics, KU Leuven, Leuven, Belgium
[d] Faculty of Economics and Business, KU Leuven, Leuven, Belgium

ARTICLE INFO

Article history:
Received 20 June 2021
Received in revised form 6 November 2021
Accepted 15 November 2021
Available online 27 November 2021

Keywords:
Feature engineering
Interpretability
Fraud detection
Credit risk

ABSTRACT

In many practical applications, such as fraud detection, credit risk modeling or medical decision making, classification models for assigning instances to a predefined set of classes are required to be both precise and interpretable. Linear modeling methods such as logistic regression are often adopted since they offer an acceptable balance between precision and interpretability. Linear methods, however, are not well equipped to handle categorical predictors with high cardinality or to exploit nonlinear relations in the data. As a solution, data preprocessing methods such as weight of evidence are typically used for transforming the predictors. The binning procedure that underlies the weight-of-evidence approach, however, has been little researched and typically relies on ad hoc or expert-driven procedures. The objective in this paper, therefore, is to propose a formalized, data-driven and powerful method. To this end, we explore the discretization of continuous variables through the binning of spline functions, which allows for capturing nonlinear effects in predictor variables and yields highly interpretable predictors that take only a small number of discrete values. Moreover, we extend the weight-of-evidence approach and propose to estimate the proportions using shrinkage estimators. Together, this method offers an improved ability to exploit both nonlinear and categorical predictors to achieve increased classification precision while maintaining the interpretability of the resulting model and decreasing the risk of overfitting. We present the results of a series of experiments in fraud detection and credit risk settings, which illustrate the effectiveness of the presented approach.

Code metadata

Permanent link to reproducible Capsule: https://doi.org/10.24433/CO.9447810.v1.

1. Introduction

Classification is a well-studied machine learning task that concerns the assignment of instances to a set of outcomes. Classification models support the optimization of managerial decision making across a variety of operational business processes. For instance, fraud detection models classify instances, such as transactions or claims, as fraudulent or nonfraudulent [1]. This allows for the efficient and effective allocation of limited inspection capacity by selecting the most suspicious cases for investigation by a human fraud analyst [2]. Credit risk models, on the other hand, assess the risk connected with providing credit to customers, and this risk can be used to construct optimal portfolios of loans or other lines of credit [3,4].

A wide variety of classification models have been proposed in the literature. These proposals range from very complex models including neural networks, support vector machines and ensemble methods to more elementary models such as logistic regression and decision trees [5]. Some of the more complex models have been shown to outperform the simpler classification techniques in various real-life classification tasks [6–10]. In industry, however, simple logistic regression currently remains among the most frequently used approaches for developing classification models across various fields of application [8,9,11–13]. Its popularity may be explained by the presence of industry regulations, e.g., the Basel regulatory framework for the banking industry, which requires the resulting model to be both interpretable [14] and accurate. Logistic regression is widely perceived as offering

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author at: Department of Mathematics, University of Antwerp, Antwerp, Belgium.
E-mail address: tim.verdonck@uantwerpen.be (T. Verdonck).

https://doi.org/10.1016/j.asoc.2021.108160
1568-4946/

the best balance between both objectives. Other possible explanations are the broad expertise and experience in using logistic regression that exists in industry, but follow-the-herd behavior and some degree of inertia and resistance to change may explain its enduring popularity. Moreover, the superior performance of the more complex models can strongly depend on the task at hand. On tabular datasets, which are commonly encountered in the context of credit scoring and healthcare analytics, they have been shown to provide only marginal performance gains [8,15].

Aside from the development of classification models and techniques for learning them, a different approach to improving the final model is to focus on pre- and post-processing. In contrast to studies on learning models and post-processing techniques [16,17], relatively few studies focus on preprocessing data. The goal of preprocessing is to optimally prepare the data (e.g., through transformation) to maximize the predictive power and out-of-sample performance, or, importantly, to improve the interpretability of the resulting model. Specifically, we identify a lack of approaches that allow us to optimally transform nonlinear patterns and categorical variables with high cardinality for incorporation in linear models to achieve an interpretable yet powerful classifier [18]. Currently, the weight-of-evidence (WOE) approach appears to be frequently used to this end, as it offers a good balance between interpretability and predictive power, and it is complementary with and similar to logistic regression [19,20]. For categorical variables with many categories, however, WOE may lead to overfitting. Moreover, WOE does not have an integrated binning approach for optimally merging categories or discretizing continuous predictors.

In this article, we present an integrated WOE-based approach for optimally transforming predictor variables, which mainly improves upon the existing WOE approach in cases with nonlinear predictor variables (continuous or ordinal) and categorical predictor variables with high cardinality. The goal of this preprocessing method is to maximize both the predictive power and interpretability of logistic regression models (and more generally, generalized linear models). The proposal is based on generalized additive models in combination with exact univariate k-means clustering and shrinkage estimation. The presented approach is experimentally evaluated; an illustration of the use of the proposed approach and an indication of its merits are provided. An open source implementation of the method is provided in the digital annex to this paper to enable peer researchers to reproduce and verify the presented results and allow practitioners to adopt the method for practical use. This paper is structured as follows. In the following section, we present the standard methodology that uses logistic regression and weight-of-evidence, and we expand upon this approach in Section 3. In Section 4, we present experimental results obtained from a fraud detection case and a credit risk case, and in Section 5, we conclude the paper and present directions for future research.

2. Background methodology

Consider a model with a binary response Y and p continuous predictors $\boldsymbol{X} = (X_1, \ldots, X_p)$. The goal is to model the conditional mean $p_{\boldsymbol{x}} = E(Y|\boldsymbol{X} = \boldsymbol{x}) = P(Y = 1|\boldsymbol{X} = \boldsymbol{x})$. The classical logistic regression model, which is part of the family of generalized linear models (GLMs) [21], assumes a linear relationship between the predictor variables and the log-odds of the event $Y = 1$. More specifically, we have

$$\log\left(\frac{p_{\boldsymbol{x}}}{1-p_{\boldsymbol{x}}}\right) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i = \beta_0 + \boldsymbol{\beta x}$$

where β_0 denotes an intercept and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ denotes a vector of model parameters. This model can be reformulated in terms of probabilities as

$$P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \frac{1}{1 + e^{-(\beta_0 + \boldsymbol{\beta x})}}.$$

The classical logistic regression model serves as a very popular benchmark for many binary classification tasks due to its ease of computation, high interpretability and solid performance. However, it also has several shortcomings, two of which we want to focus our attention on:

1. categorical variables with many categories
2. continuous variables with nonlinear effects on the log-odds

Categorical variables are often one-hot encoded (also known as "dummy encoding"), after which they can be included in the model as numerical variables. This has the drawback that a categorical variable with N categories leads to $N - 1$ variables. If N is large, this leads to considerable variability in the estimation process and usually many insignificant predictors. One way to avoid this problem is by converting the categorical variable into a continuous variable by using a weight-of-evidence transformation. The weight-of-evidence (WOE) transformation of a categorical predictor is commonly defined as follows. Suppose that we have a category j with N_j elements. Denote by P_j the number of true cases in our category and by F_j the number of false cases in our category. Additionally, let P be the total number of true cases in the data and F be the total number of false cases in the data. The WOE value of category j is then given by:

$$\log\left(\frac{P_j/P}{F_j/F}\right). \tag{1}$$

The WOE transformation usually provides an elegant solution, but since it is based on the estimation of a proportion, its variance can be high when there are categories with few observations, which is common for categorical variables with high cardinality.

Continuous variables are modeled by logistic regression as having linear effects on the log-odds of the response. While this is often reasonable, there can be variables that do not satisfy this assumption. This happens unexpectedly but sometimes by design, as illustrated in the following example. Suppose that we want to predict whether a transaction is fraudulent based on a single predictor X_t that characterizes the time at which the transaction was made (i.e., taking values within $[0, 24)$). Now suppose that we make the reasonable assumption that the influence of the time on the probability of a transaction being fraudulent is roughly continuous, and we interpret X_t being close to 24 as X_t being close to 0. Then, we would have $P(Y = 1|X_t = 0) = \lim_{T\to 24^-} P(Y = 1|X_t = T)$. In terms of log-odds, this would imply that $\beta_0 = \lim_{T\to 24^-} \beta_0 + \beta_1 T = \beta_0 + 24\beta_1$, which clearly can only be satisfied when $\beta_1 = 0$. In other words, under the assumptions above, the only relationship that can be fit is a constant relationship, which is not of much interest. This example illustrates that some variables display nonlinear relationships with the response by design.

One way to incorporate the nonlinear effects of continuous predictors on the log-odds is to use the generalized additive model (GAM, [22,23]) for logistic regression:

$$\log\left(\frac{p_{\boldsymbol{x}}}{1-p_{\boldsymbol{x}}}\right) = \beta_0 + \sum_{i=1}^{p} f_i(x_i) \tag{2}$$

where $f_1, \ldots, f_p$ are arbitrary smooth functions of the predictor variables $x_1, \ldots, x_j$. The model in Eq. (2) is very flexible, but this flexibility comes at a price. As the functions f_i can be arbitrary

smooth functions of the predictors, they can display rather unusual patterns. These factors make the model harder to interpret and hence less used in practical situations such as fraud detection, where the predictions resulting from the model may have to be explained. To improve the interpretability of the model, [24] proposed a data-driven way of binning the fitted functions f_i into a limited number of categories. Afterwards, a classical logistic regression model can be fit to the binned variable. This strategy allows for capturing nonlinear effects while greatly improving the interpretability of the model.

Throughout the remainder of the article, we make the assumption that the conditional expectation of Y can indeed be adequately modeled through a GAM in the predictor variables. This assumption entails that the nonlinear effects are sufficiently smooth functions of the predictors. Furthermore, we assume that the number of variables p is considerably smaller than the number of observations n which guarantees stability in fitting the model. In case this last assumption would not be met, one could resort to regularized GAMs and apply the proposed methodology in that setting.

3. Methodology

In the following, we describe our proposal to address the issues described in the previous section. The underlying goal is to develop a powerful predictive model while maintaining interpretability by allowing the incorporation of nonlinear effects within a GLM and by improving upon the traditional WOE-based binning process.

3.1. (Local) shrinkage of WOE

Our starting point for the treatment of categorical variables is the WOE transformation that transforms a categorical variable into continuous values. To introduce our shrinkage estimator for the WOE values, we first rewrite the definition of Eq. (1) in a different but equivalent form. More specifically, for a given categorical variable, we assign the empirical log-odds to each bin, i.e., each element in a given category j is assigned the value

$$\widehat{\mathrm{WOE}}_j = \log\left(\frac{\hat{p}_j}{1-\hat{p}_j}\right) \tag{3}$$

where $\hat{p}_j$ denotes the proportion of successes (e.g., fraudulent transactions) in category j. The equivalence with the earlier definition in Eq. (1) can be seen as follows. With the notation introduced before, we have that $\mathrm{WOE}_j = \log\left(\frac{p_j}{1-p_j}\right) = \log\left(\frac{P_j/N_j}{F_j/N_j}\right) = \log\left(\frac{P_j}{F_j}\right) = \log\left(\frac{P_j/P}{F_j/F}\right) + \log\left(\frac{P}{F}\right)$. Therefore, both values differ by only a constant, which typically does not play a role in most statistical or machine learning models. As an example, the constant disappears in the intercept of a GLM. It is worth noting that sometimes categories with $\hat{p} = 0$ or $\hat{p} = 1$ can occur, and these lead to undefined WOE values. In those cases, we can slightly adjust the WOE by introducing a small offset c with $0 < c < 1$ and replace $\hat{p} = 0$ with $\hat{p} = \frac{c}{n_j}$ and $\hat{p} = 1$ with $\hat{p} = 1 - \frac{c}{n_j}$. Note that this offset disappears as the number of observations in the category becomes large (i.e., when $n_j \to \infty$). We use $c = 0.01$ by default. In practice, categories are often merged to avoid this boundary case, but this merging introduces a certain level of arbitrariness. In particular, it raises the question as to whether all possible combinations of categories should be considered as possible merging candidates. Additionally, this technique does not use the performance or quality of the final model for evaluating which merges are most interesting. We thus prefer working with a small offset, after which we can deal with the WOE values in a rigorous way.

For a category with a small number of observations n_j, the estimation of p_j (and the corresponding WOE_j) has a high variance, often yielding unreliable estimates. This is more likely to occur in categorical variables with many levels. To address this issue, we consider the shrinkage estimation of the proportion of successes in each category j. The shrinkage estimator of a proportion is given by [25]:

$$\tilde{p}_j = (1-b_j)\hat{p}_j + b_j\hat{p}$$

where $\hat{p}$ denotes the proportion of successes calculated over all possible values of j (i.e., over all categories). We thus effectively shrink the proportion of successes towards the sample mean. The shrinkage coefficient b_j determines the amount of shrinkage: $b_j = 0$ corresponds to no shrinkage, whereas $b_j = 1$ corresponds to taking the population proportion. The value of b_j is chosen to minimize the expected mean squared error over all estimated proportions, which is given by $\mathrm{EMSE} = E_s[E_j[(\tilde{p}_j - p_j)^2|p_j]]$. The minimum is given by (provided $n_j/n < 0.5$):

$$b_j^* = \frac{v_j(1-n_j/n)}{v_j(1-2n_j/n) + v + \sigma^2}$$

where $v = \mathrm{var}(\hat{p})$ is the sampling variance of $\hat{p}$, v_j denotes the sampling variance of $\hat{p}_j$ and σ^2 equals the between-area variance (i.e., $\mathrm{var}_j(p_j)$) [25]. By plugging the shrinkage estimator into the WOE calculation, we obtain the shrinkage estimator of the WOE values:

$$\widehat{\mathrm{SWOE}}_j = \log\left(\frac{\tilde{p}_j}{1-\tilde{p}_j}\right)$$

for each category j. In the rest of the paper, we denote the WOE transformation based on the shrinkage estimation of the proportions by $\mathrm{SWOE}(\cdot)$.

In addition to the global shrinkage method described above, which shrinks proportions towards the overall proportion in the data, we consider shrinking the proportions locally. More specifically, we cluster the WOE values using the weighted k-means approach [26,27], where the weights are taken as inversely proportional to the sampling variability of the WOE values. Note that by the central limit theorem and delta method, it holds that $\sqrt{n}(g(\hat{p}) - g(p)) \xrightarrow{D} N\left(0, \frac{1}{p(1-p)}\right)$, where $g(t) = \log\left(\frac{t}{1-t}\right)$. The asymptotic variance of the WOE estimates is thus $1/(np(1-p))$. Denoting the WOE values with $z_1, \ldots, z_n$, we therefore solve the optimization problem given by

$$\hat{B}_1, \ldots, \hat{B}_K = \underset{B_1,\ldots,B_K}{\arg\min} \sum_{k=1}^{K} \sum_{i \in B_k} w_i (z_i - \bar{z}_k)^2$$

where $w_i \sim n_{j_i}\hat{p}_{j_i}(1-\hat{p}_{j_i})$ and j_i is the category of the original observation x_i. Note that these weights are small for categories with very few observations, which makes it more likely that these categories are put in the same cluster as other categories. Clustering the WOE values induces local shrinkage, since WOE values that are close together tend to end up in the same cluster and receive a WOE value that is a weighted average of the WOE values in the cluster. In addition to achieving less variability in the estimation of the WOE values, we also obtain a natural "fusing" of similar categories resulting in a categorical variable with fewer categories. This allows for easier interpretation and visualization of the effect of the categorical variable. In the rest of the paper, we denote the WOE transformation based on clustered estimation of the proportions by $\mathrm{CWOE}(\cdot)$.

To allow for nonlinear effects of the predictor variables on the log-odds, we revisit the approach of [24] and start from the generalized additive model (GAM) of Eq. (2). After fitting the GAM, the goal is to discretize the fitted spline functions into a

limited number of bins. These can then be used as inputs for a classical logistic regression model. As such, we can capture nonlinear effects while greatly improving the interpretability of the model.

Our approach differs from others in three main ways. First, we unite different binning types in one framework consisting of "constrained" and "unconstrained" binning. Both types have the same elegant objective function (with an additional constraint in the former case), which can be optimized exactly and efficiently. Second, we avoid the use of evolutionary trees for constrained binning, as they are typically slow to compute and do not guarantee a global optimum of the objective function. Finally, our framework allows for a natural inclusion of weights in both types of binning, and these are typically chosen to be inversely proportional to the variance of the estimated spline function at the observed value. This strategy avoids creating too many bins in those regions of the spline function which are supported by only a few observations.

Depending on the nature of the predictor variable, different types of binning may be desirable. We distinguish two cases:

1. **Unconstrained** binning: the value of the original feature does not play a role in the binning process.
2. **Constrained** binning: the value of the original feature imposes a monotonicity constraint on the binning process.

Let us consider an example. Suppose that x_j is a variable characterizing the age of a person making a transaction. After fitting the model in Eq. (2), we obtain a smooth function $f_j(x_j)$ that linearly influences the log-odds. Suppose that we want to create bins for this transformed variable. If we apply unconstrained binning, the binning of $f_j(X_j)$ would be independent of the value of X_j. This means that the resulting bins may combine different age groups. We could have a bin of ages $\{0-20, 80+\}$ and another bin of ages $\{21-79\}$. While this may be fine in some situations, there may also be situations where the binning process is required to be contiguous in x_j to enable a user to interpret or explain the model. This means that the categories cannot "jump" over ages. An example of such a binning result is {0–50} and {50+}. We would like to emphasize that the choice of binning is primarily a question of user preferences.

Unconstrained binning is arguably the easiest problem. Given a predictor $\boldsymbol{x} = x_1, \ldots, x_n$ where $i = 1, \ldots, n$ ranges over the observations, consider the transformed values $z_i = f(x_i)$. We want to find K disjoint bins $\hat{B}_1, \ldots, \hat{B}_K$ for the original observations $x_1, \ldots, x_n$ such that within each bin, the corresponding values of z_i are roughly homogeneous. This is a univariate clustering problem for which many approaches have been proposed. We propose to optimize the weighted k-means objective function:

$$\hat{B}_1, \ldots, \hat{B}_K = \underset{B_1,\ldots,B_K}{\arg\min} \sum_{k=1}^{K} \sum_{i \in B_k} w_i (z_i - \bar{z}_k)^2$$

where $w_i \geq 0$ are weights such that $\sum_{i=1}^{n} w_i = n$ and $\bar{z}_k$ denotes the mean of all z_i values with $i \in B_k$ (i.e., the cluster center). We choose the weights to be inversely proportional to the variance of the fitted spline function at point x_i. Once we obtain the bins $\hat{B}_1, \ldots, \hat{B}_K$, we can transform the original predictor $\boldsymbol{x} = x_1, \ldots, x_n$ to $\bar{z}_{k_1}, \ldots, \bar{z}_{k_n}$, where k_i denotes the cluster to which observation $i = 1, \ldots, n$ is assigned. Alternatively, we can include the predictor as a categorical variable with the categories equal to the cluster memberships. We choose not to do this to avoid the creation of many dummy variables.

The weighted k-means clustering problems can be solved exactly in $\mathcal{O}(n \log(n))$ time using dynamic programming. Finally, note that the k-means approach with all weights equal to 1 is equivalent to Fisher's natural breaks algorithm [28] used in [24].

The issue of choosing the number of clusters K is a challenge in cluster analysis, and a multitude of heuristic approaches exist. Among the more popular methods are the gap statistic [29] and the silhouette coefficient [30]. While these can be used in our setting, they do not take our primary goal of building a solid predictive model into account. Therefore, we adopt a hyperparameter tuning approach and determine the value of K by evaluating the quality of the resulting logistic regression model, aligning the clustering process with our overall objective. We address this issue in more detail in Section 3.2.

We now turn to the problem of constrained binning. Consider again the transformed variable $z_i = f(x_i)$. In contrast to the unconstrained binning scenario, the value of x_i now influences the clustering of the z_i values. Suppose without loss of generality that the values of x_i are ordered in the relevant order (e.g., the observed ages are listed in ascending order). We are now interested in K bins $B_1, \ldots, B_K$, which each contain disjoint subsets of $x_1, \ldots, x_n$ such that if $x_i, x_j \in B_k$ for certain $i < j \in \{1, \ldots, n\}$, then $x_l \in B_k$ for all $i \leq l \leq j$. Of course, we still want the bins to contain homogeneous values for the corresponding transformed values z_i. This problem is equivalent to fitting a step function to the set of bivariate points (x_i, z_i), i.e., we look for a piecewise-constant approximation of z_i within the clusters of x_i. This problem has been considered in many areas, including function approximation, time series analysis and cluster analysis. In the same spirit as the weighted K-means approach, we propose to optimize the weighted K-segments objective function:

$$\hat{B}_1, \ldots, \hat{B}_K = \underset{B_1,\ldots,B_K}{\arg\min} \sum_{k=1}^{K} \sum_{i \in B_k} w_i (z_i - \bar{z}_k)^2$$

which is the exact same objective as that of the weighted k-means problem, with the added constraint that the bins need to be contiguous. The weights $w_i \geq 0$ are again chosen to be inversely proportional to the variance of the fitted function at point x_i.

The k-segments clustering can be found exactly in $\mathcal{O}(n^2)$ time using dynamic programming, but an approximate $\mathcal{O}(n \log(n))$ algorithm exists [31]. Alternatively, one could use (evolutionary) regression trees to bin the z_i values. However, they are typically slower to compute and do not guarantee a global optimum of the objective function.

3.2. Model building

We now discuss how to incorporate the new techniques when building a GLM. For each continuous effect that is discretized into a step function, there is one tuning parameter in the form of the number of bins used. For categorical data, the global shrinkage estimation of the WOE values does not have additional tuning parameters, but when using clustering to achieve local shrinkage, the number of clusters is a tuning parameter. Ideally, one would optimize a performance criterion of choice over all possible combinations of the tuning parameters, but this evidently becomes computationally cumbersome when there are multiple nonlinear continuous variables and clustered categorical variables.

We propose to simplify the problem as follows. A simple AIC for univariate k-means clustering is [32] AIC $= \text{WCSS}_k + 2k$, where k equals the number of clusters and WCSS_k denotes the within-cluster sums of squares (i.e., the value of the k-means objective) when clustering into k clusters. One could use this formula to select the number of clusters for each clustering problem, but this would not take the performance of the final model into account. Therefore, we adapt this criterion by introducing a parameter that balances the strength of the fit with the number of clusters:

$$\text{WCSS}_k + \lambda k. \tag{4}$$

Table 1
Tuning strategy for the binning of the splines.

Tuning of λ_c and λ_{uc}	
Step 1	Bin the unconstrained nonlinear continuous effects using the number of bins k that yields the minimal value of the objective in Eq. (4) with $\lambda = \lambda_{uc}$.
Step 2	Bin the constrained nonlinear continuous effects using the number of bins k that yields the minimal value of the objective in Eq. (4) with $\lambda = \lambda_c$.
Step 3	Fit a GLM using the binned effects, possibly including other variables.
Step 4	Evaluate the GLM using the AIC.

Table 2
Tuning strategy for the clustering of the WOE values of categorical variables.

Tuning of λ_{cat}	
Step 1	For each categorical variable, find the number of clusters associated with the value of $\lambda = \lambda_{cat}$.
Step 2	Cluster the WOE values of all categorical variables using the appropriate number of clusters found in the previous step.
Step 3	Train the GAM using the splines for the continuous variables and the clustered WOE values for the categorical variables.
Step 4	Evaluate the GAM using the AIC.

As λ increases, we encourage the algorithm to use fewer bins or clusters. When there are only continuous variables that need to be preprocessed, we propose to use two tuning parameters, λ_c and λ_{uc}, for the constrained and unconstrained effects, respectively. Note that it is necessary to distinguish between these two effects since the constrained problem will have a naturally higher WCSS. To tune the model, we thus use the procedure outlined in Table 1.

Finally, we choose the combination of tuning parameters yielding the lowest AIC value. This procedure can be used in combination with the shrinkage estimation of the WOE values since the latter procedure does not have a tuning parameter. If the WOE values need to be clustered as well, there is one additional tuning parameter λ_{cat}. In that case, this parameter is optimized first, as the nature of an effect (linear vs. nonlinear) may change after clustering the WOE values. For each value of λ_{cat}, we thus execute the procedure outlined in Table 2, after which the value of λ_{cat} yielding the lowest AIC of the resulting GAM is retained.

Once λ_{cat} has been determined, we proceed by tuning λ_c and λ_{uc} using the previous procedure in Table 1. Note that instead of using the AIC, the tuning parameters can also be tuned using other performance criteria, such as a measure of prediction accuracy, on a validation set (if available) or through cross validation. This requires more data to be available and more computation time but is likely to better guard against the overfitting of the training data. The parameters λ_{co} and λ_{ca} yielding the lowest out-of-sample prediction errors are then retained, and the final model is fit using these values.

We now analyze the computational complexity of the whole pipeline including the tuning procedure. Suppose the data consists of n observations in p dimensions in addition to a univariate response. Furthermore, assume that the continuous variables can be split up in p_{uc} unconstrained nonlinear effects, p_c constrained nonlinear effects, and p_l linear effects. Also denote the number of categorical variables with p_{cat} so that $p = p_{uc} + p_c + p_l + p_{cat}$. Finally, We assume that the lengths of the grids for the tuning parameters are given by G_{cat}, G_{uc} and G_c.

The time complexity can now be analyzed by splitting up the procedure in 2 steps, the first being the tuning of λ_{cat} for the categorical variables, and the second the tuning of the parameters λ_{uc} and λ_c for the continuous nonlinear effects.

Step 1 requires, for each value of λ_{cat}, the preprocessing of p_{cat} variables and the fitting of one GAM on $\boldsymbol{X}$. The preprocessing requires $\mathcal{O}(n\log(n))$ time for each categorical variable due to the exact univariate k-means optimization. In total, we thus obtain $\mathcal{O}(G_{cat}\,(C_{GAM} + p_{cat}\,n\log(n)))$, where C_{GAM} denotes the computational cost of fitting a GAM to the data. Note that step 1 is only needed when the WOE values need to be binned. In case shrinkage estimation is used, there is no need for the tuning parameter λ_{cat} and the complexity becomes $\mathcal{O}(C_{GAM})$. The complexity of fitting a GAM depends on the fitting algorithm and the number of smoothing parameters, but $\mathcal{O}(np^2)$ is a reasonable assumption given a fixed number of iterations until convergence (see [33,34] for a discussion).

Step 2 requires the separate preprocessing of the constrained and unconstrained effects through spline-binning. This requires $\mathcal{O}((G_c p_c + G_{uc} p_{uc})\, n\log(n))$ time. Additionally, for each combination of λ_{uc} and λ_c, the fitting of one GLM is required, which leads to an additional $\mathcal{O}(G_c G_{uc} np^2)$ cost.

Combining the computational cost of both steps together, we obtain a total of $\mathcal{O}(G_{cat}\left(np^2 + p_{cat} n\log(n)\right)) + \mathcal{O}((G_c p_c + G_{uc} p_{uc}) n\log(n)) + \mathcal{O}(G_c G_{uc} np^2)$. If we assume the sizes of the grid to be constant for increasing n and p, and we further assume that at least one of $p_{uc}, p_c, p_l, p_{cat}$ is $\mathcal{O}(p)$ (which is a worst-case scenario), we obtain an overall complexity of $\mathcal{O}(n\log(n)p + np^2)$. While this is a manageable complexity, the constant factor may be quite high if the grids for parameter tuning are fine. That said, the optimization over a grid can be easily parallelized to allow for efficient yet precise parameter tuning.

4. Empirical results

4.1. Data

We evaluate our proposal on two datasets. The first is a dataset on fraud detection in credit card transactions completed on the east coast of the USA. The dataset consists of training and test sets with 3334 and 3335 points, respectively. For each transaction, 5 variables are recorded: `amount`, `age`, risk `category` (previously assigned by the bank), `country` and `time`. The response is a binary variable indicating fraudulent transactions, of which there are 73 in this dataset (i.e., roughly 1%). Table 3 presents an overview of the variables in the dataset, and Fig. 1 shows the histograms of the continuous variables.

As a second illustration of our proposal, we use the dataset from the 2009 Pacific–Asia Knowledge Discovery and Data Mining conference (PAKDD) competition. This dataset is about credit risk assessment for private label credit card applications. After removing the constant predictors, we are left with 40000 observations of 20 predictor variables. The response is again binary and indicates whether a credit card application is good or bad, with approximately 20% of the applications in the data being bad. The data are publicly available, in, among others, the `CostCla`

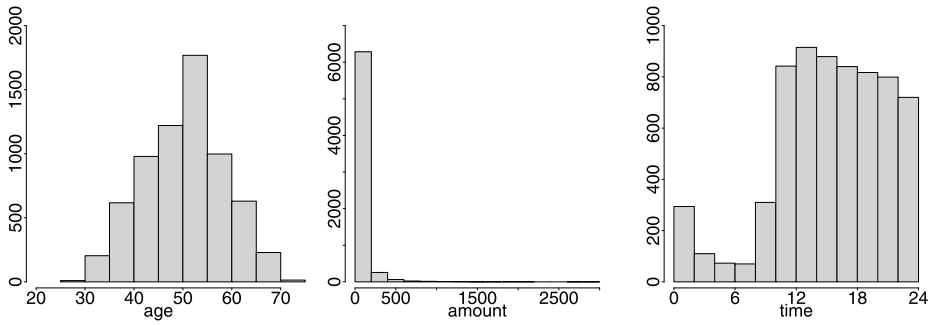

Fig. 1. Histograms of the continuous variables in the credit card fraud dataset. The age variable (left) is roughly symmetrically distributed, the amount variable is heavily right skewed, and the time variable shows few transactions between 1 and 7 a.m..

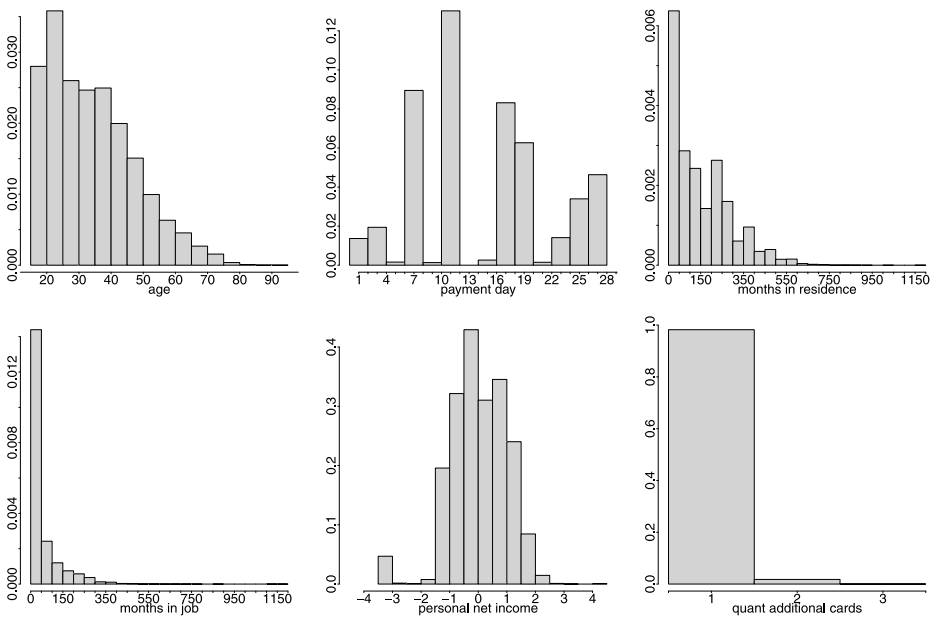

Fig. 2. Histograms of the continuous variables in the credit risk dataset. The personal net income variable is transformed using a power transformation from the Yeo–Johson family.

Table 3
Description of the variables in the fraud detection dataset.

Variable name	Description
amount	Transaction amount (USD)
age	Age of the person executing the transaction
category	Risk category of the transaction (low-medium-high)
country	Transaction destination (43 countries)
time	Time of transaction (0–24 h)

Library [35]. Of the 20 variables, there are 7 numerical and 13 categorical variables. The names of the categorical variables are listed in Table 4 together with the number of categories of each variable. As is clear from this table, there are a number of binary variables but also some variables with multiple categories, including the variable PROFESSION_CODE with 289 levels.

Fig. 2 presents the histograms of the continuous variables in the credit risk dataset, with the exception of the variable MATE_INCOME, which has over 95% zeros and does not allow for an elegant histogram representation. The personal net income variable is transformed towards normality using the Yeo–Johnson power transformation [36] fitted by weighted maximum likelihood [37].

4.2. Experimental design

To illustrate the advantages of the proposed method in several ways, we set up three experiments. The first two are conducted on the credit card fraud data and are meant to illustrate the model building process step-by-step while emphasizing the enhanced interpretability and superior results of the resulting model. The third experiment is a complete comparison of the proposed method on the credit risk data using cross validation. For our experiments, we make use of the R packages mgcv [38], Ckmeans.1d.dp [31], cellWise [39], hmeasure [40], xgboost [41], caret [42] and ROCR [43].

4.2.1. Experiment 1: the effect of spline binning on the fraud dataset

In the first experiment, we use only the continuous variables to predict fraudulent transactions. To quickly scan for the variables that may have potential nonlinear effects on the response, we fit a GAM on the continuous predictors. Fig. 3 shows the results, indicating that the amount and time variables are likely to influence the log-odds of fraud in a nonlinear way. Note that the time variable is a typical example of an inherent nonlinear effect, as discussed in Section 3.

Table 4
Categorical variables in the credit risk dataset.

Variable name	Number of categories
ID_SHOP	31
SEX	2
MARITAL_STATUS	5
FLAG_RESIDENCIAL_PHONE	2
AREA_CODE_RESIDENCIAL_PHONE	59
SHOP_RANK	3
RESIDENCE_TYPE	4
FLAG_MOTHERS_NAME	2
FLAG_FATHERS_NAME	2
FLAG_RESIDENCE_TOWN_eq_WORKING_TOWN	2
FLAG_RESIDENCE_STATE_eq_WORKING_STATE	2
PROFESSION_CODE	289
FLAG_RESIDENCIAL_ADDRESS_eq_POSTAL_ADDRESS	2

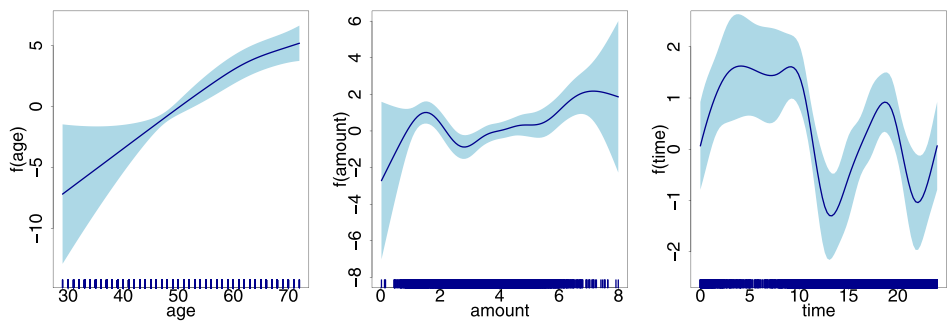

Fig. 3. Results of a classical GAM fit to the continuous predictors. The fitted splines suggest a quasi-linear effect for the age variable (left) and nonlinear effects for the amount (middle) and time (right) variables on the log-odds.

Denoting by p the probability of fraud, we train the following GAM on the training data:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \texttt{age} + f_1(\texttt{amount}) + f_2(\texttt{time}) \tag{5}$$

where f_1 is a thin-plate regression spline [44] and f_2 is a cyclic cubic regression spline [23], which captures the periodic nature of the time effect.

In the second step, the continuous effects $f_1(\texttt{amount})$ and $f_2(\texttt{time})$ are discretized (i.e., approximated by step functions) using the strategy described in Section 3.2 to obtain $\texttt{f(amount)}$ and $\texttt{f(time)}$. The amount variable is discretized using constrained binning, whereas we use unconstrained binning for the time variable.

Finally, a classical logistic regression model is fit to the transformed variables:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \texttt{age} + \beta_2 \texttt{f}_1(\texttt{amount}) + \beta_3 \texttt{f}_2(\texttt{time})$$

The results are evaluated based on different criteria. In addition to the AIC on the training set, we also evaluate the AUC, the weighted Brier score and the H-measure obtained on the test set. The AUC is the well-known area under the receiver operating curve (also equivalent to a linearly transformed Gini coefficient). The classical Brier score is the mean squared error between the predicted probabilities and observed responses, i.e., $\frac{1}{n}\sum_{i=1}^{n}(\hat{p}_i - y_i)^2$. This measure is clearly inadequate for imbalanced classification tasks, as it gives equal importance to each individual prediction. We therefore use weights that are inversely proportional to the prior probabilities: $\text{wbrier} = \frac{1}{n}\sum_{i=1}^{n} w_i(\hat{p}_i - y_i)^2$, where $w_i = \frac{1}{\pi_0} I_{y_i=0} + \frac{1}{\pi_1} I_{y_i=1}$. Note that these weights make the predictions of all fraudulent cases together as important as those of all regular transactions. The H-measure is a more recently developed alternative to the AUC that avoids dependence on the classifier and is therefore more reliable. It requires the severity ratio as an input, for which we take the recommended ratio of the class priors (π_1/π_0); see [45,46] for details.

4.2.2. Experiment 2: complete approach on the fraud dataset

In the second experiment, we consider the complete fraud dataset (including the categorical variables) with the goal of evaluating the different treatment combinations of the categorical and continuous variables. For the combination of discretized splines with the shrinkage estimation of the WOE values, we first convert the categorical variables into continuous variables using shrinkage estimators. Then, we proceed as in Experiment 1, with the difference being that the GAM now includes the transformed categorical variables:

$$\begin{aligned}\log\left(\frac{p}{1-p}\right) &= \beta_0 + \beta_1 \texttt{age} + \beta_2\, \text{SWOE}(\texttt{category}) \\ &+ \beta_3\, \text{SWOE}(\texttt{country}) + f_1(\texttt{amount}) \\ &+ f_2(\texttt{time})\end{aligned}$$

where f_1 is a thin-plate regression spline and f_2 is a cyclic cubic regression spline, which captures the periodic nature of the time effect.

For the combination of the clustered WOE values with the discretized splines, we follow the strategy outlined in Section 3.2. We thus first optimize the number of clusters for each of the categorical variables using the approach in Table 2. Afterwards, we proceed as in Experiment 1 but now with the GAM:

$$\begin{aligned}\log\left(\frac{p}{1-p}\right) &= \beta_0 + \beta_1 \texttt{age} + \beta_2\, \text{CWOE}(\texttt{category}) \\ &+ \beta_3\, \text{CWOE}(\texttt{country}) + f_1(\texttt{amount}) \\ &+ f_2(\texttt{time})\end{aligned}$$

For the evaluation, we use the same performance measures as in the previous experiment: the AIC, AUC, weighted Brier score and H-measure.

4.2.3. Experiment 3: complete approach on the credit risk dataset

In this experiment, we use the same approach as in Experiment 2 in that we compare the combinations of spline binning

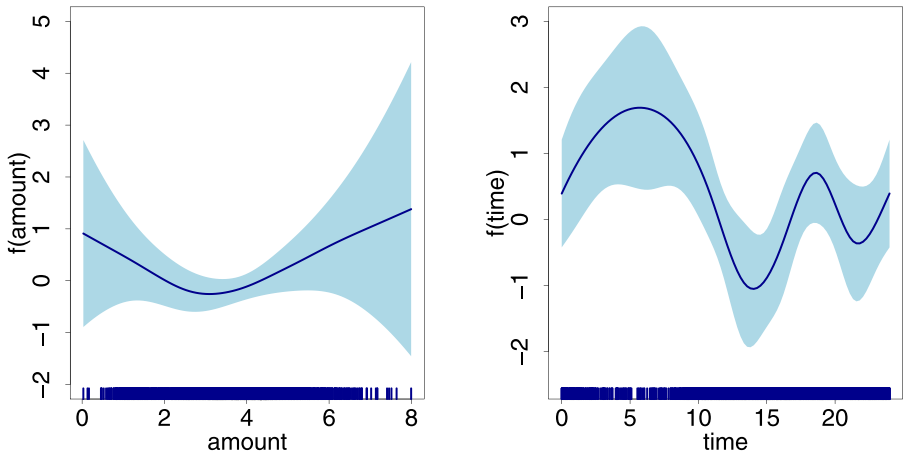

Fig. 4. The estimated spline functions of the initial GAM fit for the `amount` (left) and `time` (right) variables.

with the different treatments of categorical variables. We again use the strategy outlined in Section 3.2, including the clustering of the categorical variables as in Table 2 when cWOE is used. All of the continuous variables are fit as a binned spline in the model, with the exception of `QUANT_ADDITIONAL_CARDS`, as it is supported on a very discrete domain. The `PAYMENT_DAY` variable, which indicates the day of the month on which the eventual payments will be made, is fitted with a cyclic spline, as it is natural to expect cyclic behavior from this variable. As there is no predefined split for the training and test data, we evaluate our proposal using 10-fold cross validation and evaluate the performance of the method on each fold using the AIC, AUC, weighted Brier score and H-measure.

4.3. Results

4.3.1. Experiment 1

The initial fit of the GAM of Eq. (5) yields the estimates $\hat{\beta}_0 = -19.518$ and $\hat{\beta}_1 = 0.268$, in addition to the spline functions f_1 and f_2 shown in Fig. 4. The fitted amount effect suggests that extreme amounts (both large and small) are more likely to be fraudulent. The time effect suggests that transactions in the morning and late afternoon are more likely to be fraudulent, whereas transactions in the early afternoon and early evening are less likely to be fraudulent. The fitted GAM has an AIC of 284.495. For the out-of-sample measures, we obtain an AUC of 0.919, a weighted Brier score of 0.407 and an H-measure of 0.604. This is a reasonable performance, and we will compare it to the final model and classical GLM later.

We now discretize the fitted spline functions. We choose a maximum of $k = 10$ bins and use the selection strategy detailed in Section 3.2. This yields 7 bins for the constrained `amount` binning and 6 bins for the unconstrained binning of the `time` variable. Fig. 5 shows the original and binned effects of both variables. In the left panel, we see the amount variable discretized via a step function with 7 steps. Note that the first and last steps span a rather large interval of transaction amounts. The reason is that there are fewer observations in these regions, and the variance of the estimated spline is much larger. Therefore, due to the weighting strategy with weights inversely proportional to the variances, we obtain larger bins at the extremes of the spline. The right panel shows the time variable, which we wrap around a circle in a clock plot for the purpose of presentation. This plot visually illustrates the time windows in which transactions are more likely to be fraudulent. Note that an effect such as this could never be estimated using classical logistic regression.

Table 5
Comparison of the different models trained on the continuous predictors of the fraud detection dataset.The GLM with spline binning (SB) outperforms the other methods in the out-of-sample evaluation, whereas the classical GAM has a slightly lower AIC.

Method	AIC	AUC	wbrier	H-measure
`classical GLM`	293.656	0.896	0.438	0.549
`classical GAM`	**284.495**	0.919	0.407	0.604
`SB GLM`	286.429	**0.925**	**0.396**	**0.624**
`XGBoost`	NA	0.891	0.363	0.567

We now evaluate the performance of the obtained model using the various performance measures discussed above. The final GLM fit on the discretized splines and the original `age` variable has an AIC of 286.429. This is slightly above the AIC of the full GAM, but it is clear that the difference is rather small. Furthermore, the tables turn when considering out-of-sample performance. The proposed method yields an AUC of 0.925, a weighted Brier score of 0.396 and an H-measure of 0.624. All of these are in fact better than the corresponding performance measures of the classical GAM fit. This can be explained by the fact that the classical GAM may slightly overfit the training data. By discretizing the resulting spline functions, we gain robustness against this overfitting. Table 5 shows a comparison of the performances. We additionally add the results of the classical GLM. We see that the GLM with spline binning (SB) outperforms the classical GLM on all levels. The most significant difference is found in the H-measure, with an increase of almost 15%. As a reference, we add the performance of XGBoost (XGB) [47] to the table, which does not provide a significant improvement over the GLM-based approaches on these data.

4.3.2. Experiment 2

In the second experiment, we compare the different combinations of our proposed preprocessing techniques. The results of this comparison are presented in Table 6. Several interesting conclusions can be made from these results. First, we see that the classical GLM is vastly outperformed by any of the other methods. This is mainly due to the inclusion of 42 dummy variables for the categorical variable `country`. Second, we can see that the shrinkage estimation of the WOE values outperforms the classical WOE, regardless of whether the continuous effects are estimated using discretized splines. The clustered WOE values do not significantly outperform the classical WOE values, and their main benefit thus lies in the fact that the final model is more interpretable, since it enforces a natural reduction in the number

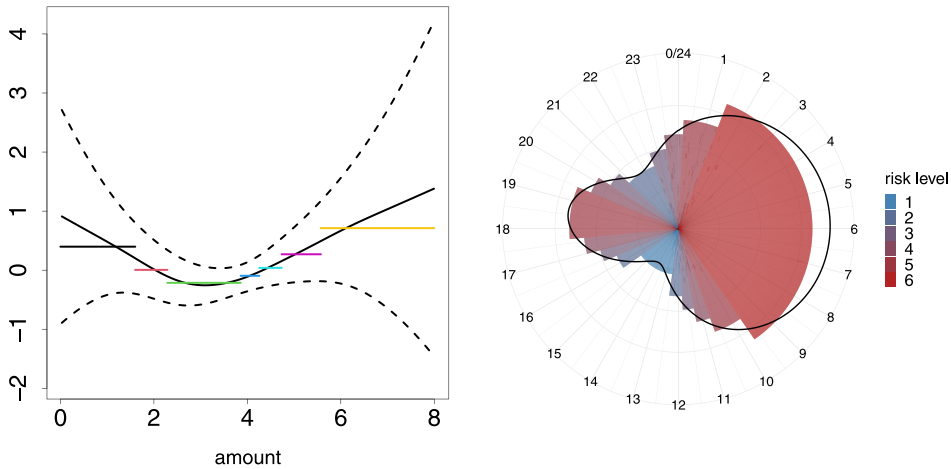

Fig. 5. The discretized spline functions of the initial GAM fit for the `amount` (left) and `time` (right) variables.

Table 6
Evaluation of the combined strategies on the credit card fraud dataset. The shrinkage estimation of the WOE values in combination with spline binning outperforms the other models. The clustered WOE values in combination with spline binning is the second best-performing model.

WOE	sWOE	cWOE	SB	AIC	AUC	wbrier	H
☐	☐	☐	☐	285	0.831	0.366	0.520
☑	☐	☐	☐	227	0.925	0.352	0.596
☐	☑	☐	☐	226	0.928	0.354	0.615
☐	☐	☑	☐	225	0.924	0.357	0.589
☑	☐	☐	☑	217	0.941	**0.335**	0.638
☐	☑	☐	☑	**216**	**0.943**	**0.336**	**0.652**
☐	☐	☑	☑	219	0.936	**0.336**	0.627
☐	☐	☐	XGB	NA	0.905	0.347	0.637

of categories within the categorical variables. Finally, we see that the discretized spline approach always improves upon the model obtained using the original continuous variables. The XGB classifier now outperforms the classical GLM but has an inferior performance to that of the GLM approach after preprocessing with WOE.

For illustrative purposes, we further analyze the model obtained using clustered WOE values and spline binning. The clustering of the categorical variables yields an optimal tuning parameter of $\lambda_{cat} = e^{-7}$. This parameter enforces a clustering of the `country` variable into 12 bins (down from 42 categories), whereas the `category` variable is left untouched with its original 3 categories. Fig. 6 shows the binned `country` variable with 12 different levels. It turns out that transactions going to Europe are generally connected to lower probabilities of fraud, with the exception being receivers in Greece (and the UK to a lesser extent). The highest risk is associated with national transactions and those to Canada and Mexico. International transactions to Australia, China, South Africa and Chile have neutral risk levels.

The GAM fit with the optimal value of λ_{cat} no longer displays a nonlinear effect for the `amount` variable, as was the case in Experiment 1. This means that the inclusion of the categorical variables resolves the nonlinearity issue for this variable, and we can treat it as a linear effect. The `time` variable, however, still displays a nonlinear relationship with the response, as shown in Fig. 7.

Discretizing the continuous effect of the `time` variable yields 3 bins. The result of this binning step is shown in Fig. 8. It is clear that the transactions made in the morning or early evening are more likely to be fraudulent than the transactions around noon or late in the evening. The coefficients of the final model

Table 7
Coefficients of the final model.

	Estimate	P-value
(Intercept)	−12.55	0.00
`amount`	0.19	0.19
`age`	0.27	0.00
CWOE(`category`)	0.64	0.01
CWOE(`country`)	0.90	0.00
`f(time)`	1.88	0.00

Table 8
Evaluation of the combined strategies on the credit risk dataset. The shrinkage estimation of the WOE values in combination with spline binning outperforms the other models.

WOE	sWOE	cWOE	SB	AIC	AUC	wbrier	H
☑	☐	☐	☐	33304.29	0.6693	0.3105	0.1043
☐	☑	☐	☐	33413.02	0.6701	0.3104	0.1056
☐	☐	☑	☐	33305.30	0.6692	0.3105	0.1045
☑	☐	☐	☑	**33184.43**	0.6732	0.3087	0.1093
☐	☑	☐	☑	33291.17	**0.6746**	**0.3083**	**0.1112**
☐	☐	☑	☑	33185.57	0.6733	0.3086	0.1096
☐	☐	☐	XGB	NA	0.6546	0.3168	0.0886

are presented in Table 7, which suggests that all predictors have significant contributions to the final model, with the exception of the `amount` variable.

4.3.3. Experiment 3

The results for the final experiment are summarized in Table 8. It is clear that the absolute differences are not as pronounced as those in the previous example. This is not very surprising, as a significant number of predictor variables that carry a lot of signal are either binary or enter the model linearly, and in both cases, the effect of the proposed approach is limited. Nevertheless, all the differences are statistically significant, as verified by the Wilcoxon rank test [48], which yields p-values between 0.002 and 0.036 for testing the performance of sWOE + SB against the alternatives in terms of the AUC, wbrier and H-measure. These differences can produce significant cost savings in practical business settings. As in the previous example, the XGBoost classifier does not seem to improve upon a GLM-based approach for these data.

4.4. Discussion

The results of the experiments above lead us to several conclusions. First, in regard to the estimation of WOE values, estimating the proportions using the shrinkage estimator seems to improve

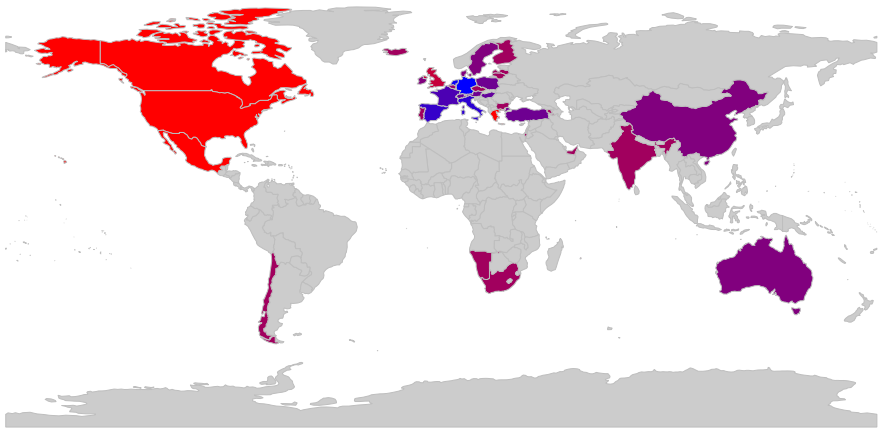

Fig. 6. The `country` variable reduced to 12 categories instead of the original 42.

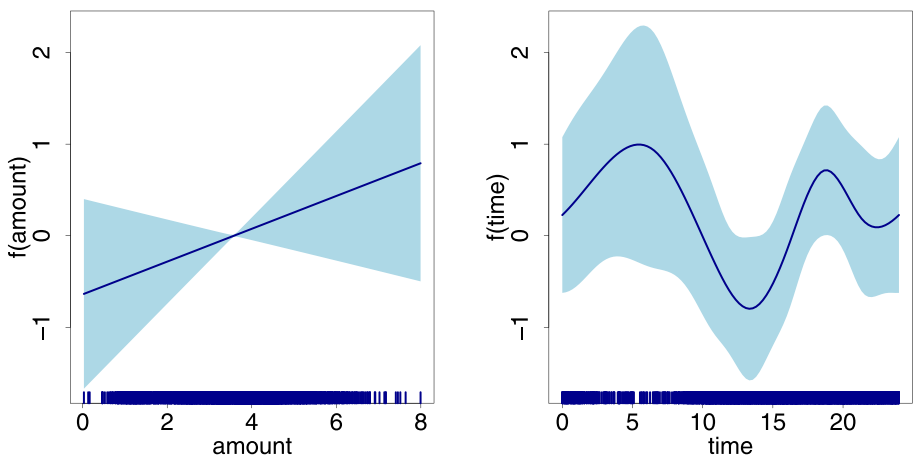

Fig. 7. The estimated spline functions of the initial GAM fit when all variables are included in the model. The `amount` variable (left) no longer displays a nonlinear effect on the response variable, as was the case for the model with only continuous variables.

the out-of-sample performance of the resulting model. Second, clustering the WOE values does not generally yield a substantial improvement over the regular WOE values but has the advantage of fusing the categorical variables into a variable with fewer categories, thereby improving the interpretability of the model. Finally, the use of binned splines on the continuous variables significantly improves the out-of-sample performance of the model. Additionally, one could argue that this also leads to improved interpretability, as the continuous variables are reduced to a select number of discrete values. Note that the advantage of using binned splines may not be significant if there are no important nonlinear effects in the set of predictor variables.

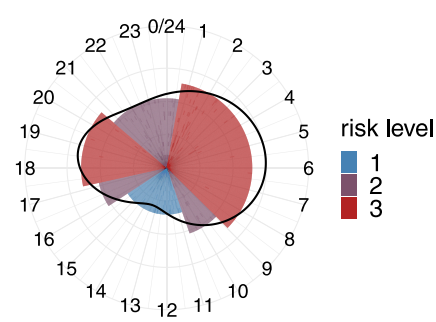

Fig. 8. The effect of the binning time on the final model.

5. Conclusion

We propose and study two advanced techniques for preprocessing data before applying regression. The first method considers the treatment of WOE values, which we propose to estimate using shrinkage estimators for the proportions. Alternatively, the original WOE values can be clustered for improved interpretability. Second, we study the discretization of continuous variables through the binning of spline functions. This allows for capturing nonlinear effects in predictor variables and yields highly interpretable predictors that take only a small number of discrete values.

Through three different experiments on a fraud detection dataset, we illustrate the advantages of using these advanced preprocessing techniques. In particular, the out-of-sample performance of the model is improved using the binned spline treatment on the continuous variables. Additionally, the WOE values obtained based on shrinkage estimation of the proportions also increase the out-of-sample performance of the resulting model. The clustering of WOE values shows improved interpretability but no clear improvement in predictive performance.

When it comes to the limitations of the proposed method, three points need mentioning. The first is that it should be possible to adequately model the conditional expectation of the response given the predictors should be appropriately modeled through a generalized additive model. Since this is the starting point of the modeling pipeline, it is a rather obvious yet important

limitation. The second is that the computational cost gets quite high when there are many nonlinear continuous effects. As the number of such effects gets higher, GAMs become less and less suitable for modeling. The final limitation is that of risk of overfitting. Whenever GAMs are used, there is the risk of overfitting to the training data, and smoothing parameter selection should be carefully executed. However, there exist reliable automatic routines for this. Further research could address the combination of the two strategies for categorical variables by using the classical WOE values as inputs for a GAM. This combined method would be able to capture the nonlinear effects of the WOE values on the response. However, due to the nature of WOE in logistic regression (which implies a linear WOE effect on the response), it is not clear that this would yield an improvement over the current method. Another line of research could investigate a more precise approximation of the spline functions in the GAM. For example, one could use a piecewise linear approximation instead of a step function, which would still be easy to interpret but more flexible to work with. Finally, the shrinkage estimation of the proportions could be combined with clustering, i.e., one could first compute WOE values based on shrinkage estimation and then cluster the resulting values in a number of bins.

Software availability

An implementation of the proposed pipeline as well as a script reproducing the results in the paper can be found in the GitHub repository https://github.com/JakobRaymaekers/WOE2.0 and on the website of our research unit, https://wis.kuleuven.be/statdatascience/robust/software.

CRediT authorship contribution statement

Jakob Raymaekers: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft. **Wouter Verbeke:** Conceptualization, Methodology, Formal analysis, Investigation, Validation, Writing – review & editing. **Tim Verdonck:** Conceptualization, Methodology, Formal analysis, Investigation, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors gratefully acknowledge the financial support from the BASF Research Chair on Robust Predictive Analytics, the BNP Paribas Fortis Research Chair in Fraud Analytics at KU Leuven, Belgium and the Internal Funds KU Leuven, Belgium under grant C16/15/068. The funders had no role in the study design, data collection and analysis process, the decision to publish, or the preparation of the manuscript.

References

[1] J. Vanhoeyveld, D. Martens, B. Peeters, Value-added tax fraud detection with scalable anomaly detection techniques, Appl. Soft Comput. 86 (2020) 105895, http://dx.doi.org/10.1016/j.asoc.2019.105895.

[2] B. Baesens, V. Van Vlasselaer, W. Verbeke, Fraud Analytics using Descriptive, Predictive, and Social Network Techniques: A Guide to Data Science for Fraud Detection, John Wiley & Sons, 2015.

[3] B. Baesens, D. Roesch, H. Scheule, Credit Risk Analytics: Measurement Techniques, Applications, and Examples in SAS, John Wiley & Sons, 2016.

[4] C. Bluhm, L. Overbeck, C. Wagner, Introduction to Credit Risk Modeling, Crc Press, 2016.

[5] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer Science & Business Media, 2009.

[6] Y.-C. Chang, K.-H. Chang, G.-J. Wu, Application of extreme gradient boosting trees in the construction of credit risk assessment models for financial institutions, Appl. Soft Comput. 73 (2018) 914–920, http://dx.doi.org/10.1016/j.asoc.2018.09.029.

[7] H.-Y. Shi, K.-T. Lee, H.-H. Lee, W.-H. Ho, D.-P. Sun, J.-J. Wang, C.-C. Chiu, Comparison of artificial neural network and logistic regression models for predicting in-hospital mortality after primary liver cancer surgery, PLoS One 7 (4) (2012) e35781, http://dx.doi.org/10.1371/journal.pone.0035781.

[8] S. Lessmann, B. Baesens, H.-V. Seow, L.C. Thomas, Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research, European J. Oper. Res. 247 (1) (2015) 124–136.

[9] B.R. Gunnarsson, S. vanden Broucke, B. Baesens, M. Óskarsdóttir, W. Lemahieu, Deep learning for credit scoring: Do or don't? European J. Oper. Res. (2021) http://dx.doi.org/10.1016/j.ejor.2021.03.006.

[10] M. Óskarsdóttir, C. Bravo, C. Sarraute, J. Vanthienen, B. Baesens, The value of big data for credit scoring: Enhancing financial inclusion using mobile phone data and social network analytics, Appl. Soft Comput. 74 (2019) 26–39, http://dx.doi.org/10.1016/j.asoc.2018.10.004.

[11] S.Y. Sohn, D.H. Kim, J.H. Yoon, Technology credit scoring model with fuzzy logistic regression, Appl. Soft Comput. 43 (2016) 150–158, http://dx.doi.org/10.1016/j.asoc.2016.02.025.

[12] B. Baesens, T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, J. Vanthienen, Benchmarking state-of-the-art classification algorithms for credit scoring, J. Oper. Res. Soc. 54 (6) (2003) 627–635.

[13] X. Dastile, T. Celik, M. Potsane, Statistical and machine learning models in credit scoring: A systematic literature survey, Appl. Soft Comput. 91 (2020) 106263.

[14] D. Martens, J. Vanthienen, W. Verbeke, B. Baesens, Performance of classification models from a user perspective, Decis. Support Syst. 51 (4) (2011) 782–793.

[15] A. Rajkomar, E. Oren, K. Chen, A.M. Dai, N. Hajaj, M. Hardt, P.J. Liu, X. Liu, J. Marcus, M. Sun, et al., Scalable and accurate deep learning with electronic health records, NPJ Digit. Med. 1 (1) (2018) 18.

[16] W. Verbeke, D. Martens, B. Baesens, Rulem: Rule learning with monotonicity constraints for ordinal classification, Appl. Soft Comput. 60 (2017) 858–873.

[17] M. Herasymovych, K. Märka, O. Lukason, Using reinforcement learning to optimize the acceptance threshold of a credit scoring model, Appl. Soft Comput. 84 (2019) 105697, http://dx.doi.org/10.1016/j.asoc.2019.105697.

[18] J. Moeyersoms, D. Martens, Including high-cardinality attributes in predictive models: A case study in churn prediction in the energy sector, Decis. Support Syst. 72 (2015) 72–81.

[19] E.P. Smith, I. Lipkovich, K. Ye, Weight-of-evidence (WOE): Quantitative estimation of probability of impairment for individual and multiple lines of evidence, Hum. Ecol. Risk Assess. 8 (7) (2002) 1585–1596.

[20] R. Anderson, The Credit Scoring Toolkit: theory and Practice for Retail Credit Risk Management and Decision Automation, Oxford University Press, 2007.

[21] J.A. Nelder, R.W. Wedderburn, Generalized linear models, J. R. Stat. Soc. Ser. A Gen. 135 (3) (1972) 370–384.

[22] T. Hastie, R. Tibshirani, Generalized additive models: Some applications, J. Amer. Statist. Assoc. 82 (398) (1987) 371–386.

[23] S.N. Wood, Generalized Additive Models: An Introduction with R, second ed., Chapman and Hall/CRC, 2017.

[24] R. Henckaerts, K. Antonio, M. Clijsters, R. Verbelen, A data driven binning strategy for the construction of insurance tariff classes, Scand. Actuar. J. 2018 (8) (2018) 681–705, http://dx.doi.org/10.1080/03461238.2018.1429300.

[25] N.T. Longford, Multivariate shrinkage estimation of small area means and proportions, J. Roy. Statist. Soc. Ser. A 162 (2) (1999) 227–245.

[26] 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: Statistics, University of California Press, Berkeley, Calif., 1967, pp. 281–297.

[27] S. Lloyd, Least squares quantization in PCM, IEEE Trans. Inform. Theory 28 (2) (1982) 129–137.

[28] W.D. Fisher, On grouping for maximum homogeneity, J. Amer. Statist. Assoc. 53 (284) (1958) 789–798.

[29] R. Tibshirani, G. Walther, T. Hastie, Estimating the number of clusters in a data set via the gap statistic, J. R. Stat. Soc. Ser. B Stat. Methodol. 63 (2) (2001) 411–423, http://dx.doi.org/10.1111/1467-9868.00293.

[30] P.J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65, http://dx.doi.org/10.1016/0377-0427(87)90125-7.
[31] H. Wang, M. Song, Ckmeans.1d.dp: Optimal k-means clustering in one dimension by dynamic programming, R J. 3 (2) (2011) 29.
[32] S.A. Ramsey, S.L. Klemm, D.E. Zak, K.A. Kennedy, V. Thorsson, B. Li, M. Gilchrist, E.S. Gold, C.D. Johnson, V. Litvak, et al., Uncovering a macrophage transcriptional program by integrating evidence from motif scanning and expression dynamics, PLoS Comput. Biol. 4 (3) (2008) e1000021.
[33] S.N. Wood, Z. Li, G. Shaddick, N.H. Augustin, Generalized additive models for gigadata: Modeling the UK black smoke network daily data, J. Amer. Statist. Assoc. 112 (519) (2017) 1199–1210.
[34] Z. Li, S.N. Wood, Faster model matrix crossproducts for large generalized linear models with discretized covariates, Stat. Comput. 30 (1) (2020) 19–25.
[35] A. Correa Bahnsen, CostSensitiveClassification library in Python, 2015, http://dx.doi.org/10.5281/zenodo.17789.
[36] I.-K. Yeo, R.A. Johnson, A new family of power transformations to improve normality or symmetry, Biometrika 87 (4) (2000) 954–959.
[37] J. Raymaekers, P.J. Rousseeuw, Transforming variables to central normality, Mach. Learn. (2021) 1–23.
[38] S. Wood, mgcv: Mixed GAM computation vehicle with GCV/AIC/REML smoothness estimation, 2012, URL https://CRAN.R-project.org/package=mgcv. R package version 1.8-36.
[39] J. Raymaekers, P. Rousseeuw, cellWise: Analyzing data with cellwise outliers, 2021, R package version 2.2.5.
[40] C. Anagnostopoulos, D.J. Hand, hmeasure: The H-measure and other scalar classification performance metrics, 2019, URL https://CRAN.R-project.org/package=hmeasure. r package version 1.0-2.
[41] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, M. Li, J. Xie, M. Lin, Y. Geng, Y. Li, XGBoost: Extreme gradient boosting, 2021, URL https://CRAN.R-project.org/package=xgboost. R package version 1.3.2.1.
[42] M. Kuhn, cAret: classification and regression training, 2020, URL https://CRAN.R-project.org/package=caret. R package version 6.0-86.
[43] T. Sing, O. Sander, N. Beerenwinkel, T. Lengauer, ROCR: Visualizing classifier performance in R, Bioinformatics 21 (20) (2005) 7881.
[44] S.N. Wood, Thin-plate regression splines, J. R. Stat. Soc. Ser. B Stat. Methodol. 65 (1) (2003) 95–114.
[45] D.J. Hand, Measuring classifier performance: A coherent alternative to the area under the ROC curve, Mach. Learn. 77 (1) (2009) 103–123.
[46] D.J. Hand, Evaluating diagnostic tests: The area under the ROC curve and the balance of errors, Stat. Med. 29 (14) (2010) 1502–1510.
[47] T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 785–794, http://dx.doi.org/10.1145/2939672.2939785.
[48] F. Wilcoxon, Individual comparisons by ranking methods, in: Breakthroughs in Statistics, Springer, 1992, pp. 196–202.