Spline-rule ensemble classifiers with structured sparsity regularization for interpretable customer churn modeling

Koen W. de Bock, Arno de Caigny

Spline-Rule Ensemble Classifiers with Structured Sparsity Regularization for Interpretable Customer Churn Modeling

Koen W. De Bock[1] and Arno De Caigny[2,3]

[1] Audencia Business School, 8 Route de la Jonelière, F-44312, Nantes, France

[2] IESEG School of Management, 3 Rue de la Digue, F-59000, Lille, France

[3] LEM-CNRS 9221, 3 Rue de la Digue, F-59000, Lille, France

E-mail addresses: kdebock@audencia.com (Koen W. De Bock), a.de-caigny@ieseg.fr (Arno De Caigny)

Corresponding author: Koen W. De Bock, Audencia Business School, 8 Route de la Jonelière, F-44312, Nantes, France. E-mail address: kdebock@audencia.com, Tel.: +33 2 40 37 34 34

This article is published in ***Decision Support Sytems.***

Please cite as:

De Bock, K. W., & De Caigny, A. (2021). Spline-rule ensemble classifiers with structured sparsity regularization for interpretable customer churn modeling. Decision Support Systems, Volume 150, November 2021, 113523

Article doi: https://doi.org/10.1016/j.dss.2021.113523

Spline-Rule Ensemble Classifiers with Structured Sparsity Regularization for Interpretable Customer Churn Modeling

Abstract

An important business domain that relies heavily on advanced statistical- and machine learning algorithms to support operational decision-making is customer retention management. Customer churn prediction is a crucial tool to support customer retention. It allows an early identification of customers who are at risk to abandon the company and provides the ability to gain insights into why customers are at risk. Hence, customer churn prediction models should complement predictive performance with model insights. Inspired by their ability to reconcile strong predictive performance and interpretability, this study introduces rule ensembles and their extension, spline-rule ensembles, as a promising family of classification algorithms to the customer churn prediction domain. Spline-rule ensembles combine the flexibility of a tree-based ensemble classifier with the simplicity of regression analysis. They do, however, neglect the relatedness between potentially conflicting model components which can introduce unnecessary complexity in the models and compromises model interpretability. To tackle this issue, a novel algorithmic extension, spline-rule ensembles with sparse group lasso regularization (SRE-SGL) is proposed to enhance interpretability through structured regularization. Experiments on fourteen real-world customer churn data sets in different industries (i) demonstrate the superior predictive performance of spline-rule ensembles with sparse group lasso over a set well yet powerful benchmark methods in terms of AUC and top decile lift; (ii) show that spline-rule ensembles with sparse group lasso regularization significantly outperform conventional rule ensembles whilst performing at least as well as conventional spline-rule ensembles; and (iii) illustrate the interpretable nature of a spline-rule ensemble model and the advantage of structured regularization in SRE-SGL by means of a case study on customer churn prediction for a telecommunications company.

Keywords: customer churn prediction; predictive analytics; spline-rule ensemble; interpretable data science; sparse group lasso; regularized regression

1 Introduction

An important application of data science is to drive and to support data-driven decision making. Many decision makers are convinced that the use of customer-data capabilities allows to gain an unbeatable competitive advantage [1]. Therefore, modern companies have developed the analytical and technological capabilities that enable collection, storage and analysis of data. An important business domain that relies heavily on advanced statistical - and machine learning algorithms to support operational decision making is customer retention management [2]. Customer churn prediction (CCP) is of crucial importance for managing customer retention as a tool to identify customers who are at risk to abandon the company and to better understand *why* customers are at risk [3]. In line with these managerial objectives of CCP models, previous research in CCP focused both on predictive performance (i.e. detecting who is at risk) [4,5] and interpretability (i.e. understanding why a customer is at risk) [6]. Accuracy in CCP is generally pursued due to its immediate impact on campaign profitability [7]. Model interpretability is crucial to facilitate management buy-in and organizational acceptance, to deliver insights into the drivers of churn and loyalty and consequently, to provide venues for formulating strategies to remedy customer churn and promote loyalty [8,9].

Algorithms that combine good predictive performance and interpretable output, such as decision trees (DT) or logistic regression (LR), are preferred in CCP [2,10]. Ensemble learners can achieve higher predictive performance, but often lack on the interpretability criterion [4]. A notable exception are *rule ensembles* (RE), a technique that is designed to combine the merits of ensemble learners with a high degree of interpretability [11]. Like many other ensemble learners, rule ensembles first generate a set of decision trees. However, unlike other ensemble learners, trees are decomposed into rules and only a dense set of the rules derived from these trees is retained though the application of lasso regression. The initial variables are also added to the lasso regression in the form of *linear basis functions* (i.e., variable transformations) to better account for linear variable effects. Rule ensembles thus combine *terms* rather than member classifiers. The simple nature of the constituent terms that form the model and their selection through lasso regression result in an easily interpretable model. Recently, *spline-rule ensembles* (SRE) are presented as an extension to rule ensembles that complement rules and linear terms

with single-term spline functions in order to better accommodate univariate, nonlinear relationships between the dependent variable and individual explanatory variables [12].

Whilst the promise of competitive predictive performance and model interpretability has attracted attention in several domains such as bioinformatics and computer science [e.g. 13,14], applications of rule ensembles in management, and more specifically, decision support in business, remain scarce to date. In an application of corporate bankruptcy prediction, SRE demonstrated superior performance over conventional RE whilst the added value of the integration of spline functions was demonstrated [12]. Despite their promising traits, other applications of RE and SRE in business decision-making problems are very scarce and to the best of our knowledge RE and SRE have not been empirically assessed for predicting customer churn thus far. This study's primary objective is to evaluate and compare both model architectures in the domain of CCP.

RE and SRE rely on lasso regression, which does not consider relatedness that exists between covariates. This is, nevertheless, very important to consider, because the building blocks of SRE (i.e. splines, linear base functions and rules) can share a dependence on the same variables, which can cause the model to become unnecessarily complex. Imagine for example the impact on a model's ease of interpretation if a variable enters the model in three terms: a linear base function, a spline and a rule. In such a case, an analyst would face difficulties to assess the isolated effect of that variable on the churn probability. These issues are aggravated when conflicting parameter estimate signs emerge. To tackle these issues and significantly improve the interpretability of SRE, the second objective of this study is to propose a new algorithm entitled *spline-rule ensembles with sparse group lasso regularization* (SRE-SGL). SRE-SGL groups rule, spline and linear terms according to the variables upon which they depend by applying a straightforward indexing function. This term grouping is followed by sparse group lasso (SGL) regularization [15] that accommodates this group structure by enforcing regularization between as well as within term groups. As such, the co-occurrence of terms that depend on the same variable or variable set is discouraged and the complexity of the resulting model is reduced in comparison to a conventional SRE model.

The contributions of this paper are the following: (i) RE and SRE are evaluated and compared in the field of CCP and their ability to reconcile accuracy and model interpretability is assessed; and (ii) SRE-SGL, extending spline-rule ensembles with sparse group lasso regularization, is introduced as a natural extension of generic RE and SRE that simplifies model interpretation. To assess and compare predictive performance of RE, SRE and the new SRE-SGL, as well as a set of benchmark algorithms, experiments are conducted on a large set of 14 data sets containing real-world customer churn data sets in various sectors to compare RE and its extensions with a set of benchmark algorithms in terms of predictive performance. The added value offered by SRE-SGL in comparison to RE and SRE in terms of model interpretability is illustrated using an in-depth case study.

This paper is structured as follows. In the next section related research is discussed. This involves three subsections: Section 2.1 discusses the concept of interpretability in data science. Section 2.2 discusses prior literature in customer churn prediction that focusses on the trade-off between accuracy and interpretability. Section 2.3 introduces rule-based ensemble classifiers and their applications. Section 3 presents the methodology. Section 4 handles the data and the experimental design. The results of our large benchmark experiment and a case study to demonstrate the interpretability of SRE-SGL are discussed in section 5. The study's conclusions, limitations and areas for future research are presented in section 6.

2 Related Literature

2.1 Interpretability in Data Science

Interpretability is an important topic in data science and various approaches have been proposed for explaining model predictions [16,17]. Interpretability cannot be described in a pure mathematical formula, and depends on human interpretation. Hence, interpretability can be defined as the degree to which humans can understand the cause of a decision [18,19]. As the ability to understand the cause of a decision depends on the observer, interpretability is a subjective topic. Nevertheless, it is an important dimension to consider for model evaluation to ensure that predictions are unbiased, sensitive information

is protected, the reliability and robustness of the model is checked and that humans can trust the model [20].

Approaches to explain model predictions vary in scope and flexibility [17]. The scope indicates the level of explanations and can either be on the global or on the instance level. Global explanations give an insight in the model's predictions over all observations and for all possible variables' values. Instance-level explanations, on the other hand, are specific for a single prediction and help to understand why a certain instance received a specific prediction. Flexibility indicates whether the approach is specific to the model or model-agnostic. Flexibility is linked to the way interpretability is achieved. Intrinsic interpretable models achieve interpretability by restricting the complexity of the machine learning algorithm and their interpretability is thus often model-specific. The model can also be analyzed after training using so called post-hoc methods, which are often model-agnostic approaches.

Our approach focuses on global, model-specific interpretability. The output of SRE-SGL is intrinsically interpretable, which allow to interpret the model's output directly as demonstrated in the case study in section 5.2.

2.2 Interpretable Customer Churn Prediction

CCP models serve a dual purpose to decision makers; detecting customers who are at risk of churning and helping to understand why customers are at risk of churning [3]. Therefore, CCP models require not only high predictive performance but also interpretable output. In this section, we review literature in the CCP domain that focuses on churn prediction modeling as tool for better decision making.

The predictive performance of CCP models is a well-researched topic because of its importance for decision makers to detect which customers are at risk of churning. There are many strategies to improve the predictive performance of CCP models such as intelligent data preprocessing [21], data augmentation [22] or by the choice of algorithm [2]. Given the motivation of the focal study, we focus on the latter. Researchers have experimented with a wide range of algorithms in extensive benchmarking studies [4]. Such studies focused on the algorithms' ability to discriminate between churning – and non-

churning customers. Logistic regression is the standard benchmark algorithm in CCP because of its ability to produce decent and robust results [21,23]. More complex algorithms, however, frequently perform significantly better in terms of predictive performance [2,4,22,24]. The results in large benchmarking studies demonstrate that especially ensembles, such as random forests, perform well [4]. Despite the beneficial traits in terms of predictive performance, interpretability of ensembles remains an issue which causes that they are not always the preferred option.

CCP models should be interpretable in order to assist decision makers in managing customer retention. Recent studies in CCP explicitly acknowledge the importance of interpretability of predictive models. Martens et al. [10] propose a complete framework to assess the overall performance of classification models from a user perspective in terms of accuracy, interpretability and justifiability. In their analysis, interpretability is based on the output type and output size. They state that some output types, such as rules or linear ones, and smaller output sizes are intuitively easier to understand for humans.

A first strategy to obtain interpretable models in CCP is by making non-interpretable output of so called "black-box models" more interpretable through additional analyses. On the one hand, several model-agnostic interpretation techniques exist that to reveal the magnitude and nature of the effect that variables exert on a model's predictions. Notable examples are permutation-based feature importance scores, and partial dependence functions and plots. Both techniques have witnessed widespread adoption in CCP literature. On the other hand, transparent surrogate classifiers can be created to complement, or replace, opaque models. An example of this approach is rule extraction. For example, Verbeke et al. [25] experimented with new rule induction techniques, which induce accurate as well as interpretable classification rule-sets. Farquad et al. [26] propose a hybrid approach to render interpretable rule-based output for a support vector machine model. A drawback of such methods is that they only approximate the original model. The development of interpretable models is a second strategy. Miguéis et al. [27] introduce multivariate adaptive regression splines (MARS [28]) to customer churn prediction and highlight its ability to uncover nonlinear effects. Coussement et al. [29] introduced Generalized Additive Models (GAM) as a highly interpretable model in CCP. Several extensions of GAM have been presented

that improved the predictive performance while maintaining its interpretability [30–32]. Other interpretable models depend at least partly on a tree-based structure. Qi et al. [33] introduce ADTreesLogit, a model that integrates the advantage of ADTrees in the logistic regression model, to improve the predictive accuracy and interpretability of existing churn prediction models. De Caigny et al. [2] introduce the logit leaf model, a highly interpretable hybrid model based on decision trees and logistic regression that delivers actionable insights.

A final strategy involves imposing the interpretability criterion in the feature engineering. Backiel et al. [34] demonstrate how interpretable features can be extracted out of call records by using social network analysis. The use of these network features can improve the performance over local features while remaining highly interpretable. Verbraken et al. [35] stress the importance of compact networks derived from a Bayesian network classifier for the model interpretability. In the telecommunication industry, Lima et al. [36] show how domain knowledge can be incorporated in the data mining process for churn prediction.

The proposed SRE-SGL fits perfectly in the interpretable customer churn prediction literature, because SRE-SGL combines excellent predictive performance, associated with ensemble learning, and interpretability. The SRE-SGL algorithm has two main strategies to ensure interpretability. First, it returns inherently comprehensible output that can be directly analyzed by decision makers. Second, the model automatically introduces new, potentially insightful features through the combination of splines, linear base functions and rules. Such features can shed new light in the understanding of what is driving customer churn. To the best of our knowledge, RE and its extensions have never been used in CCP.

2.3 Rule-Based Ensemble Classifiers and Applications

Ensemble classification prescribes the training of multiple base learners, or member classifiers into one model and the use of a fusion rule to aggregate their individual outputs into an overall prediction [37]. The most well-known methods differ in the strategy deployed to transform the training data set into member training data for each base learner. Examples include iterative weighing of instances in adaboost [38], bootstrap sampling in bagging [39] , combining bagging and random feature selection at the node level in random forests [40], and feature extraction in rotation forests [41]. While the subject

of the algorithm choice for generating an ensemble's base learners is widely investigated, decision trees are still the most popular and the default option in the aforementioned ensemble strategies.

This study builds on previous work that has investigated the merits of deploying decision *rules* as base learners in ensemble classifiers. Decision rules can be interpreted as simple classifiers that take the form of logical expressions: if [conditions] then [decision]. The earliest surfacing of rule-based ensemble learning is, to the best of our knowledge, the SLIPPER algorithm [42] that uses boosting to create an ensemble of decision rules. Subsequently, *Rule ensembles* were proposed by Friedman and Popescu [43] to denote a class of ensemble learners for classification and regression that derive *rules* from decision trees and use them as base learners in a combination scheme based on regularized regression. A related approach was presented by Błaszczyński et al. [44] and Dembczyński, Kotłowski and Słowiński [45] who generate rules directly and use a different loss criterion. Since then, several variations, extensions and applications of rule ensembles have been presented, for example in cancer classification [46], sensor fault classification [47], analysis of start-up performance [48] and streetscape satisfaction [49]. The proposed SRE-SGL in this study extends the approach described in [12], in which rule ensembles were applied in the field of bankruptcy prediction and an extension, spline-rule ensembles, demonstrated a significant improvement in predictive performance over conventional rule ensembles.

3 Methodology

3.1 Rule and Spline-Rule Ensembles

In contrast to many well-known ensemble learners that combine decision trees, RE [43] initiates by deriving *rules* from decision trees and use them as base learners in a supervised, linear combination scheme. Consider a data set D with an input vector *X* summarizing *n* instances on p features $x_k; k = 1$ to p and an outcome vector *Y*. Specifically, RE derive rules $r_j(\mathrm{x})$; *j*=1 to q from a set of decision trees trained on *X* and *Y* for all internal and terminal nodes within every tree (interior and terminal). A rule $r_j(\mathrm{x})$ is the product of the indicator functions that define whether input vector *x* meets certain criteria defined on one or more variables:

$$r_j(x) = \prod_{s_{jk} \neq s_k} I(x_k \in s_{jk}) \qquad (1)$$

where s_{jk} represents a range or subset of values of variable x_k and S_k denotes the *full* range or set of values of this variable. Variables upon which a rule $r_j(\mathrm{x})$ depends (i.e., for which $s_{jk} \neq S_k$) are called *defining* variables. The rules are complemented by *linear basis functions* $l(x_k)$; $k = 1, \ldots, p$ which denote variables x_k subsequently subjected to winsorization and normalization as defined by:

$$l(x_k) = \frac{0.4 * win(x_k)}{sd(win(x_k))} \qquad (2)$$

with $win(x_k) = \min(\delta_k^+, \max(\delta_k^-, x_k))$ denoting the winsorized version of variable x_k and where δ_k^- and δ_k^+ specify the β^{th} and $(1-\beta)^{\text{th}}$ percentiles of x_k[1].

The final model takes the form of a linear regularized lasso-regression [50] applied to outcome vector Y and an intermediate term matrix T that contains values for $p+q$ terms $t(x)$: p variable transformations and q binary rule outcomes for all instances in a training data set.

$$\underset{\beta}{arg\,min} \frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1 \qquad (3)$$

Where shrinkage parameter λ controls sparsity: increasing values decrease the proportion of non-zero parameter estimates. $\|.\|_2$ and $\|.\|_1$ are the ℓ^2 and ℓ^1 vector norms, respectively.

Several metrics allow an identification of the relative importance and nature of relationships of the terms selected in a rule ensemble model. The first are term coefficients, i.e. the regression parameter estimates that indicate the influence a term has upon the logit transformation of the probability to churn. These deliver insight into whether a term influences churn positively or negatively, and to which extent. Furthermore, *rule support* figures are specific to rule terms and indicate for which percentage of instances in the training data set a rule applies. Finally, *term importance* measures reflect the relative importance of the terms in the model and are obtained through

[1] The constant of 0.4 is chosen so that linear terms in a subsequent regularized lasso-regression (equation (3)) receive the same a priori influence as a *typical* rule. Specifically, this constant reflects the average standard deviation that characterizes a population of rules characterized by a support that follows a uniform distribution; $\mathrm{supp}\left(\mathrm{r_j(x)}\right) \sim U(0,1)$. We kindly refer the reader to Friedman & Popescu [11] for more details.

$$TI(t) = \begin{cases} |\beta_t| \sqrt{supp\left(r_j(x)\right)\left(1 - supp\left(r_j(x)\right)\right)} & \text{if } t \text{ is a rule term; } t = r_j(x) \\ |\beta_t| . sd\left(l(x_k)\right) & \text{if } t \text{ is a linear term; } t = l(x_k) \\ |\beta_t| . sd\left(s_g(x_g)\right) & \text{if } t \text{ is a spline term; } t = s_g(x_g) \end{cases} \quad (4)$$

where $s\text{upp}\left(r_j(x)\right)$ represents the rule support for rule $r_j(x)$ [11]. Hence, $TI(t)$ represents the absolute value of the regression coefficient of a standardized term t.

Due to their nature, rule ensembles balance model flexibility and interpretability, which are classifier qualities that often conflict. Model interpretability in rule ensembles stems from the process of creating simplified, easily understandable base learners, while the regularization enforces model sparsity as many terms receive a parameter estimate equal to 0. Besides linear variable effects, variable interactions are naturally accommodated through the inclusion of rules. Moreover, these rules allow an identification of non-linear effects of individual variables on an outcome variable. However, this is only possible in an indirect manner when multiple rules, defined on the same variable are selected simultaneously.

Spline-rule ensembles aim for a more direct support of non-linear effects. To this end, a third term class was introduced in [12]: smooth functions, and in particular *penalized cubic regression splines* [51] of individual continuous variable. Penalized cubic regression splines determine a set of *v* knots $\xi_1, \xi_2, \ldots, \xi_v$ over a variable's range and estimate a function that is built up of cubic polynomials between every pair of adjacent knots. As such, they allow to model a non-linear relation between a variable and the customer churn probability. Specifically, $s(x)$ (the cubic regression spline function) takes the form

$$s(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 h(x, \xi_1) + \cdots + \beta_{v+3} h(x, \xi_v)$$

$$\text{with } h(x, \xi) = \begin{cases} (x - \xi)^3 \text{ if } x > \xi \\ 0 \text{ otherwise} \end{cases}. \quad (5)$$

Minimizing

$$\sum_{i=1}^{n} (y_i - s(x_i))^2 - \rho \int (s''(x))^2 dx \quad (6)$$

allows a determination of values for $\beta_1, \beta_2, \ldots, \beta_{v+3}$, $\xi_1, \xi_2, \ldots, \xi_v$ as well as ρ which represents a smoothing parameter, i.e. a penalty term that is required to penalize excessive curvature in the function.

The inclusion of penalized cubic regression spline functions $s(x)$ for all continuous variables $x_1, \dots, x_u$ $(u \leq p)$ in term matrix T in equation (3) updates the lasso regularization that estimates the final model. Note that in [12] experiments compared lasso regularization to ridge regression and elastic net regularization by generalizing equation (3). Results of a model variant comparison in the field of bankruptcy prediction [12] demonstrated no significant differences. Hence, in the current study these variants are not investigated further.

3.2 Sparse-Group Lasso (SGL) Regularized Regression

Rule and spline-rule ensembles rely on lasso regression to perform ensemble selection and improve interpretability through shrinkage. A limitation of lasso regression is that it does not take into account relatedness (a group structure) that exists between covariates. Variations of lasso regression enable *structured* regularization. Specifically, the *group lasso* [52] and *sparse-group lasso* (SGL) [15] allow variable *grouping*. In the case of the former, sparsity is enforced on the group level so that all variables within a selected group receive non-zero parameter estimates when their group is selected and 0 otherwise. In the case of SGL, a dual goal of sparsity is pursued: both at the between-group as the within-group level. In other words, the regression attempts to shrink the model to as few group as possible, and to as few variables within selected groups as possible. The SGL takes the following form:

$$\underset{\beta}{arg\,min}\ \frac{1}{2n}\left\|y - \sum_{o=1}^{m} T^{(o)}\beta^{(o)}\right\|_2^2 + (1-\alpha)\lambda \sum_{o=1}^{m} \sqrt{p_o}\left\|\beta^{(o)}\right\|_2 + \alpha\lambda\|\beta\|_1 \qquad (7)$$

In which m is the number of variable groups, $T^{(o)}$ is the partial term matrix reduced to variables that belong to group o, p_o is the number of variables in group o; and shrinkage is controlled by parameters λ (the shrinkage parameter) and α ($0 \leq \alpha \leq 1$), the mixing parameter that controls the trade-off between between- and within-group level regularization.

3.3 Spline-Rule Ensembles with SGL Regularization (SRE-SGL)

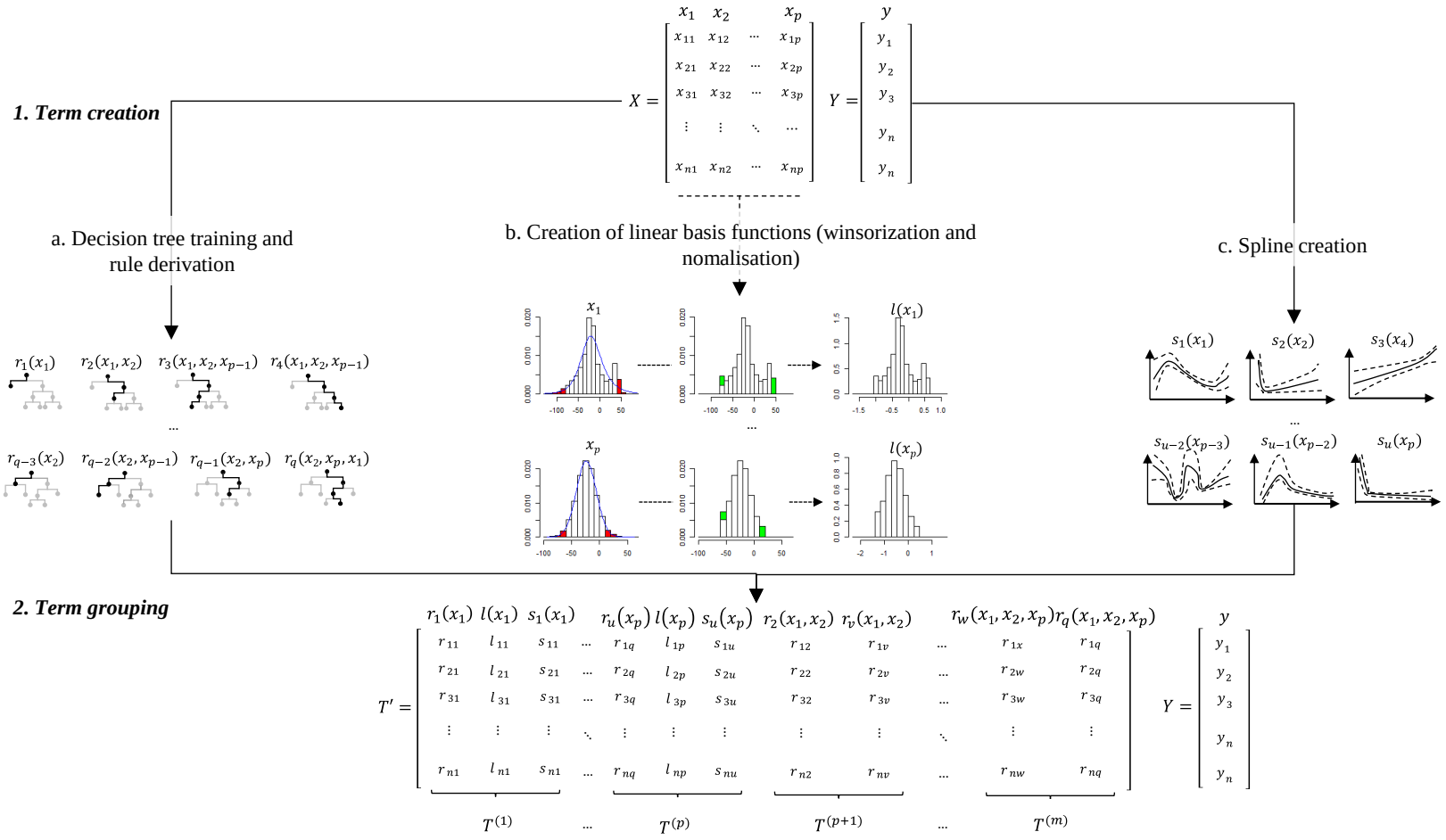
A notable disadvantage of lasso regression is that it enforces shrinkage through a loss function that does not necessarily avoid a simultaneous selection of multiple terms that rely on the same underlying variable(s). Model interpretability becomes more challenging when a variable x_k enters a model simultaneously as a linear basis function $l(x_k)$, a spline $s_k(x_k)$ and a (univariate) rule $r_l(x_k)$. Likewise,

the occurrence of similar multivariate rules that share identical defining variables complicates interpretation. These issues are aggravated when conflicting parameter estimate signs emerge. To tackle these issues and significantly improve the interpretability of spline-rule ensembles we define SRE-SGL as spline-rule ensembles with term grouping and structured regularization using SGL. SRE-SGL allows a decision maker to obtain a more interpretable model by leveraging relatedness between splines, linear basis functions and rules in term matrix *T* that share a dependence on the same variables.

Figure 1 graphically depicts the core mechanisms of SRE-SGL. The training process comprises of three stages. The first stage is identical to regular spline-rule ensembles and involves the derivation of linear basis functions, tree rules and penalized cubic regression splines that will serve as candidate ensemble members. The second stage, proper to SRE-SGL, involves *term grouping*. This involves the grouping of terms in term matrix *T* according to the variables upon which they depend. Specifically, we propose the following indexing function:

$$tg(t(\mathrm{x}_s)) := w\colon x_s = s_w \in S \qquad (8)$$

Where *t* is a term in term matrix *T*, x_s is the set of variables upon which t depends; *S* is the indexed set of unique defining term variable sets that identifies $|S| = m$ groups and s_w is the w^{th} element of *S*. For example, a multivariate term such as a rule $r_v(x_1, x_3)$ that contains conditions on variables x_1 and x_3 would contribute the set $\{x_1, x_3\}$ to *S* while univariate terms such as smoothing spline $s_2(x_2)$ and linear basis function $l(x_2)$ contributes singleton $\{x_2\}$ to *S*. $tg(t(\mathrm{x}_s))$ assigns a unique grouping index to every unique set of defining variables that emerges in the terms of matrix *T*. This encourages the term selection to choose between alternative terms that are defined on identical variables or variable sets. In the case of univariate terms, shrinkage involves selection within term sets (such as $T^{(1)}$ and $T^{(p)}$ in Figure 1) consisting of univariate rules, linear basis functions, splines or any subset of these, each dependent on the same variable. In the case of multivariate terms (i.e., rules with multiple conditions) this involves shrinkage within term sets (such as $T^{(p+1)}$ and $T^{(m)}$ in Figure 1) consisting of rules that share the same sets of defining variables.



$$\underset{\beta}{arg\,min}\ \frac{1}{2n}\left\|y-\sum_{o=1}^{m}T^{(o)}\beta^{(o)}\right\|_2^2+(1-\alpha)\lambda\sum_{o=1}^{m}\sqrt{p_o}\left\|\beta^{(o)}\right\|_2+\alpha\lambda\|\beta\|_1$$

Figure 1: Visual representation of SRE-SGL model training stages

The final component is a logistic regression with SGL regularization applied to the grouped term matrix $\mathrm{T}' = \left(T^{(1)}, T^{(2)}, \ldots, T^{(m)}\right)$ and outcome vector *Y* as identified by equation (7).

4 Experimental Validation

4.1 Data Sets

The experimental validation of SRE-SGL involves two dimensions: predictive accuracy and interpretability. For the former, the performance of SRE-SGL is compared to a set of benchmark algorithms over fourteen real-world customer churn data sets. Table 1 presents the most important characteristics of these data sets such as the industry, number of observations, number of attributes and churn incidence. Most of the data sets are proprietary and were obtained through exclusive company collaborations which limits the level of detail that can be disclosed. Therefore, the dimension of interpretability is assessed by means of a case study on publicly available data set *ds9* in Section 5.2.

Data set	Industry	# observations	# attributes	% churn	Source
Ds1	Financial Services	631,627	>100	2.53%	European Financial Services provider
Ds2	Financial Services	602,575	>100	3.16%	European Financial Services provider
Ds3	Financial Services	573,895	>100	2.57%	European Financial Services provider
Ds4	Newspaper	427,833	>100	11.14%	European newspaper company
Ds5	Financial Services	398,087	>100	4.50%	European Financial Services provider
Ds6	Financial Services	316,578	>100	6.45%	European Financial Services provider
Ds7	Financial services	117,808	>100	3.55%	European Financial Services provider
Ds8	Financial Services	102,279	>100	5.99%	European Financial Services provider
Ds9	Telecom	71,047	87	29.00%	Duke[1]
Ds10	Telecom	50,000	>100	7.34%	European telecom operator
Ds11	Telecom	47,761	43	3.69%	European telecom operator
Ds12	Retail	32,371	47	25.15%	European supermarket retailer
Ds13	Energy	20,000	33	10.00%	European energy company
Ds14	Retail	3,827	16	28.14%	European DIY retailer

[1] Center for Customer Relationship Management Duke University, February 2014. URL: http://www.fuqua.duke.edu/centers/ccrm

Table 1: Data set characteristics: data set identifier, industry, number of observations, number of attributes, customer churn percentage and source

4.2 Experimental Set-Up

First, to assess the predictive performance of SRE-SGL, a comparison is made to a set of seven benchmark algorithms: two closely related algorithms upon which it builds: conventional RE and SRE; and five algorithms that are characterized by a widespread adoption by practitioners, due to both high interpretability and strong predictive performance, on the one hand, and frequent adoption as benchmark algorithms in prior churn prediction literature on the other: regularized logistic regression, a CART

decision tree [53], random forest [40] a generalized additive logistic regression model (GAM) [54], and, finally, a multivariate adaptive regression splines (MARS) model [28]. The regularized logistic regression models are implemented with elastic net regularization. The GAM takes the form of a semi-parametric logistic regression: a binary outcome variable is predicted using a combination of splines (for continuous variables) and linear terms (for dummy variables).

Data pre-processing can have an important impact on the predictive performance of classifiers in customer churn prediction [21]. In this study, however, the impact of different data pre-processing techniques on the predictive performance of the classifiers is not one of the research objectives. Therefore, all preprocessing steps related to handling missing values, categorical variables, outliers, class imbalance and variable selection are equal for all algorithms and chosen in line with previous benchmark studies in CCP [2,4], which help to keep the study and presentation of results lean. First, missing values are imputed with the median and dummy variables are created to flag instances that are imputed for a certain variable [21]. Next, categorical variables are dummy-encoded into a number (the number of categories minus one) of binary variables that indicate the presence or absence of a particular characteristic. Since high cardinality does not pose an issue in our datasets, we did not rely on strategies to reduce the number of categories to a manageable size, such as coarse classification using hierarchical agglomerative clustering with Euclidian distance [4,55]. Then, outliers, defined as unusual values that are more than three standard deviations from the variable's mean, are transformed using winsorization. Class imbalance, a result of the number of churners being much lower than the number of non-churners, is handled by undersampling the majority class, i.e. non-churning customers, to the same level as the churners [22]. Finally, Fisher score selection is applied as an input selection procedure to reduce the dimensionality of the initial feature space to twenty [4]. This is justified because a classifier often yields equal, or even better, predictive performance on a small set of highly predictive variables than on an exhaustive set of mainly redundant variables.

A fair comparison of classifier performance requires a strategy to tune hyperparameters whilst reducing the variability in results due to sampling. To these ends, a 5x3 cross-validation experimental design, nowadays common in CCP literature [2,22,56], is deployed. This procedure involves a stratified

split of the data set in three equal parts. In each fold, one of these data parts serves as test data sample used to determine a classifier's predictive accuracy. The other two parts serve as training and validation data samples for training and evaluating a number of alternative classifier configurations by varying their hyperparameters. The configuration that corresponds to the best performance on the validation sample is chosen to train a final model on a stacked data sample obtained by combining the training and validation samples. Three estimates of predictive performance are thus obtained per fold, and this procedure is repeated five times. All classifiers are thus trained, validated and tested on exactly the same data samples.

Hyperparameter settings of the algorithms are optimized from broad ranges of values, similar as in previous CCP studies [2,4,21]. Appendix B provides an overview of the optimized hyperparameters and their candidate values for all considered algorithms. Following the approach in [12], SRE and SRE-SGL deploy penalized cubic regression splines to estimate spline terms[2]. The GAM benchmark model is configured with penalized cubic regression splines with shrinkage. Note that cubic regression splines in SRE, SRE-SGL and GAM depend on smoothness parameter ρ which is internally optimized using the generalized cross-validation (GCV) criterion [57,58] during spline estimation.

To assess the predictive performance of SRE-SGL relative to the benchmark algorithms, a statistical framework based on the non-parametric Friedman test is used as described by Demšar [59]. As 6 algorithms and 14 data sets are considered in the focal experiment, the Friedman statistic is defined as:

$$\chi_F^2 = \frac{12*14}{6(6+1)}\left[\sum_a AR_a^2 - \frac{6(6+1)^2}{4}\right] \qquad (9)$$

where AR_a denotes the average rank of the performance measures of an algorithm a= 1,2,…,6 over our 14 data sets. The Friedman test is assumed to be distributed according to χ_F^2 with *k-1* degrees of freedom under the null hypothesis that states that the results of all algorithms do not differ and thus the ranks AR_a^2 should be equal. Only if the null-hypothesis is rejected, SRE-SGL is pairwise compared with the benchmark algorithms using the Holm post-hoc test [60]. Predictive performance is measured in

[2]Experiments with alternative spline estimation methods in SRE and SRE-SGL did not reveal significant differences with respect to predictive performance. In particular, we compared penalized cubic regression splines to penalized cubic regression splines with shrinkage, penalized thin plate regression splines, P-splines and penalized thin plate regression splines with shrinkage. Detailed results of these experiments and statistical tests are available in Appendix C.

terms of area under the receiver operating characteristic curve (AUC) and top decile lift (TDL), both commonly reported in churn prediction literature.

The second part of the experimental validation of SRE-SGL involves the dimension of interpretability, assessed through an in-depth case study on one specific data set (*ds9* in Table 1). Note that for reasons of consistency, the data set is preprocessed and variables are selected as described above. However, deviating from the 5x3-fold cross-validation deployed for comparing classifiers' predictive performance, all models and derived insights reported in this section are based on a unique fold, i.e. a single data split of the Cell2cell data set.

Interpretability of RE, SRE and SRE-SGL models is in first instance assessed by understanding selected model terms, as well as the metrics that allow an identification of the relative importance and nature of relationships of the terms selected in a rule ensemble model presented in Section 3.1: coefficient estimates, term importance measures and rule support values for rule terms. Since model terms and regularization procedures vary between RE, SRE and SRE-SGL, substantial differences amongst resulting models can be reasonably expected. Therefore, our comparison of SRE-SGL to RE and SRE models involves two additional analyses that enable a comparison of aggregated variable importance and effects. Our aim is to verify to what extent the introduction of a more stringent SGL-regularization in SRE-SGL alters model effects in comparison to RE and SRE. First, since variables might emerge in multiple terms in (spline-) rule ensemble models and regularization does not prevent their simultaneous selection, the models' term importance measures are insufficient First, v*ariable importance measures* $\mathrm{VI}(x_k)$ allow to assess the relative importance of individual variables x_k in a model. This is usually of great importance to decision makers. They are obtained through:

$$VI(x_k) = \sum_{x_k \in \mathrm{x}_j} \frac{TI(t_j(\mathrm{x}_j))}{|\mathrm{x}_j|} \qquad (10)$$

which expresses the sum of term importances in which variable x_k occurs, each term divided by the cardinality of x_j, the set of variables on which it depends. Hence, higher values are awarded to variables appearing (i) more frequently and (ii) in more influential terms than others. Second, we deploy *partial dependence functions* to identify the aggregated effects that variables or variable pairs exert in RE, SRE

and SRE-SGL models. Partial dependence functions identify the isolated effect of one or more variables in a predictive model $F(\mathrm{x})$ by taking into account an averaged effect taken over the other variables [61]:

$$\hat{F}_s(x_s) = \frac{1}{n}\sum_{i=1}^{n} F(x_s, x_{i\setminus s}) \qquad (11)$$

where x_s is the set of variables of interest, *n* is, in this context, the number of customers in the data set while $\mathrm{x}_{i\setminus s}$ represents the values of customer *i* for all variables *not* occurring in variable set x_s. When x_s consists of one variable, the nature of the relationship between a single variable and the log odds of customer churn is revealed. Hence, partial dependence functions constitute a popular instrument to reveal variable effects in predictive customer scoring [e.g. 62]. Our analysis reports the corresponding partial dependence *plots*. Moreover, when x_s consists of multiple variables, equation (12) provides the basis for a quantity to analyze the presence and strength of interaction effects. Specifically, the strength of the interaction effect between variables x_j and x_k [61] can be expressed as

$$H_{jk}^2 = \sum_{i=1}^{n}\left[\hat{F}_{jk}(x_{ij}, x_{ik}) - \hat{F}_j(x_{ij}) - \hat{F}_k(x_{ik})\right]^2 / \sum_{i=1}^{n}\hat{F}_{jk}^2(x_{ij}, x_{ik}). \qquad (12)$$

5 Results

5.1 Predictive Performance Benchmark

This section presents the results of our experiment in which the predictive performance of SRE-SGL is compared with 7 benchmark algorithms over 14 CCP data sets originating from different industries. Appendix D presents the average cross-validated results, in terms of AUC and TDL respectively, for these fourteen data sets. The standard deviations over the different runs are indicated between brackets and the best performing algorithm is underlined for every data set.

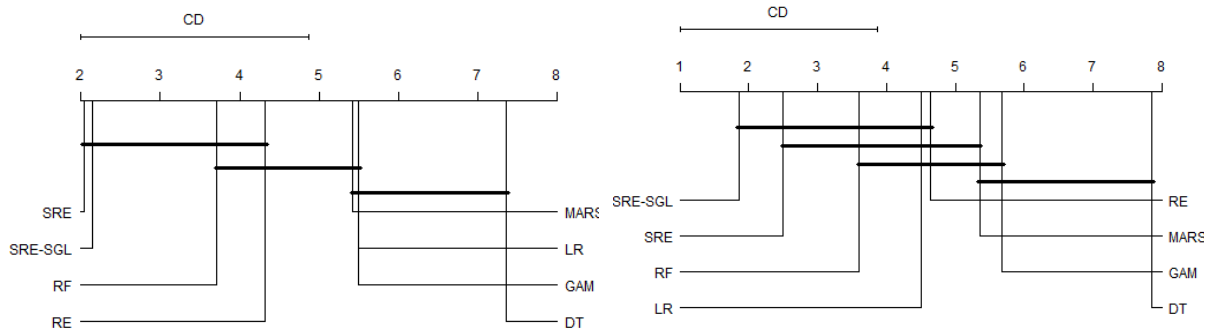
These results serve as input to determine the average ranks of the classifiers, required for the Friedman test. Table 2 displays the average ranks for the different algorithms over the 14 data sets for both AUC and TDL. A lower average rank indicates better performance. The Friedman statistic is approximately chi-squared distributed with 7 degrees of freedom and equals 55.22 (*p*-value = 0.00) based on AUC ranks and 58.84 (*p*-value= 0.00) based on TDL ranks, which indicates that there are significant differences in terms of ranks. In the post-hoc analysis, SRE-SGL serves as our control

algorithm. The adjusted *p*-values, based on the Holm post-hoc test, are given between brackets in Table 2. SRE-SGL has the lowest rank (i.e. best predictive performance) evaluated with TDL and second-best performance based on AUC. The results indicate that SRE-SGL performs always at least as well as the best performing algorithm in our benchmark. Compared with conventional RE and SRE, the results indicate that SRE-SGL performs significantly better than RE for both performance measures. There are no significant differences between SRE-SGL and SRE, indicating that reduction in model complexity of the compact SRE-SGL model over the extensive SRE model does not negatively impact the predictive performance. In comparison to traditional benchmark algorithms in CCP (i.e. DT, LR, RF, MARS and GAM), the SRE-SGL model demonstrates superior predictive performance in our benchmark study.

		Metric	
Algorithm role	**Algorithm**	**AUC**	**TDL**
Control	Spline-rule ensemble with sparse group lasso (SRE-SGL)	2.143	**1.857**
Benchmarks	Decision tree (DT)	*7.357**** (0.000)	7.857*** *(0.000)*
	Regularized logistic regression (LR)	*5.500**** *(0.002)*	*4.500*** *(0.013)*
	Random forest (RF)	3.714 (0.179)	3.607 (0.118)
	Rule ensembles (RE)	*4.321** *(0.056)*	*4.643*** *(0.011)*
	Spline-rule ensemble (SRE)	**2.036** (0.908)	2.500 (0.489)
	Generalized additive model (GAM)	*5.500**** *(0.002)*	*5.697**** *(0.000)*
	Multivariate additive regression splines (MARS)	*5.429**** *(0.002)*	*5.357**** *(0.001)*

$^{***},^{**},^{*}$ Lower average ranks indicate better performance. The best performing algorithm is indicated in bold. Indicates significance on 99%, 95%, 90% level respectively. Significant differences are indicated in *italic*. The adjusted p-value for Holm post-hoc test is shown between brackets.

Table 2: Average classifier ranks across data sets for different performance measures



(a) AUC (b) TDL

Figure 2: Critical difference plots for AUC (subplot (a)) and TDL (subplot (b))

Results show that SRE-SGL achieves significantly better predictive performance than LR,DT, MARS and GAM in terms of AUC, and it significantly outperforms DT, LR, MARS and GAM in terms

of TDL. Figure 2 summarizes the post-hoc test results visually by means of critical difference plots [63] for AUC and TDL.

5.2 Model Interpretability: A Case Study

In this section, the SRE-SGL model is assessed in terms of its ability to deliver model interpretability. To this end, a case study focusing on customer churn prediction in a telecom setting using the Cell2cell data set (*ds9* in Table 1) is presented. This public data set is well documented and has been used in previous customer churn studies [4]. The three objectives of this case study are the following: (i) to illustrate how the spline-rule ensemble model with SGL regularization results in an interpretable model, (ii) to demonstrate how SRE-SGL offers a higher degree of interpretability in comparison to rule ensemble and spline-rule ensemble models with lasso regularization and (iii), to analyze whether the introduction of structured regularization in SRE-SGL substantially changes the role of variables in the model in contrast to conventional rule and spline-rule ensembles by investigating variable importance and isolated model variable effects. Table 3 provides an overview of the 20 selected variables through applying Fisher-score selection.

Variable label	Definition	Mean	SD
callwait	Mean number of waiting calls	-0.0210	0.9759
changem	% change in minutes of use	-0.0168	0.9992
changem_M	Dummy that indicates whether *changem* (% change in minutes of use) is imputed	0.0071	0.0842
creditde	Low credit rating –de	0.1190	0.3238
custcare	Mean number of customer care calls	-0.0304	0.9684
directas	Mean number of director-assisted calls	-0.0089	0.9888
eqpdays	Number of days of the current equipment	0.0461	1.0045
incalls	Mean number of inbound voice calls	-0.0264	0.9842
models	Number of models issued	-0.0187	0.9849
mou	Mean monthly minutes of use	-0.0306	0.9838
mou_M	Dummy that indicates whether *mou* is imputed	0.0029	0.0539
opeakvce	Mean number of in and out off-peak voice call	-0.0244	0.9859
outcalls	Mean number of outbound voice calls	-0.0261	0.9844
phones	Number of handsets issued	-0.0198	0.9869
recchrge	Mean total recurring charge	-0.0335	0.9914
retcalls	Number of calls previously made to the retention team	0.0252	1.0596
revenue	Mean monthly revenue	-0.0151	0.9882
revenue_M	Dummy that indicates whether *revenue* is imputed	0.0029	0.0539
setprcm	Missing data on handset price	0.5742	0.4945
webcap	Handset is web-capable	0.8908	0.3119

Table 3: Overview of selected variables in Cell2cell data set (*ds9*). Descriptive statistics (mean and standard deviation) are provided for preprocessed training data.

5.2.1 SRE-SGL Model Interpretation

The most direct way of gaining insights into a classifier's functioning is an interpretation of the model itself. In contrast to alternative homogenous ensemble methods, rule ensembles and spline-rule ensembles deliver a facilitated model interpretability thanks to three elements: (i) the nature of candidate ensemble members (i.e., rules, linear terms and splines), (ii) their simple linear combination and (iii) the shrinkage resulting from the selection procedure to which they are submitted. SRE-SGL delivers shrinkage through sparse group lasso regularization and thus enables a more intelligent selection of competing terms through structured sparsity regularization.

Term index	Type	Term or rule specification	Coefficient	Rule support	Term importance
1	Linear term	*retcalls*	0.2828	-	100
2	Linear term	*revenue*	0.1982	-	65.3561
3	Spline	s(*changem*)	0.5164	-	40.7220
4	Spline	s(e*qpdays*)	0.2917	-	32.0586
5	Rule	*eqpdays* ≥ -0.2954 *recchrge* < 1.3767	0.1295	0.5529	21.4745
6	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2996	-0.1242	0.3901	20.2173
7	Linear term	*creditde*	-0.1805	-	19.5004
8	Linear term	*recchrge*	-0.0581	-	19.2098
9	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2954	-0.1033	0.3918	16.8194
10	Spline	s(*mou*)	0.2993	-	13.7562
11	Rule	*eqpdays* ≥ -0.2954 *recchrge* < 1.6328	0.0768	0.5640	12.7105
12	Rule	*eqpdays* ≥ -0.2954 *recchrge* < 1.1742	0.0565	0.5483	9.3771
13	Linear term	*setprcm*	-0.0445	-	7.3422
14	Spline	s(*incalls*)	0.1465	-	4.5656
15	Linear term	*changem_M*	0.1578	-	4.4317
16	Spline	s(*recchrge*)	0.0588	-	1.2510
17	Spline	s(*directas*)	0.0634	-	0.9684
18	Spline	s(*callwait*)	0.0470	-	0.3502
19	Linear term	*webcap*	-0.0029	-	0.3051
20	Spline	s(*custcare*)	0.0219	-	0.2444
21	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2913	-0.0005	0.3947	0.0868

Table 4: The Cell2cell SRE-SGL model: terms, term types, rule conditions, coefficients, rule support and term importance. Terms are sorted on the basis of their importance.

Table 4 shows the selected terms of the SRE-SGL model while Figure 3 visualizes the penalized cubic regression splines selected by the model. Table 4 also provides coefficient estimates, term

importance measures and rule support values for rule terms. Values of term importance measures $TI(t)$ are rescaled so that the most important term receives a value of 100.

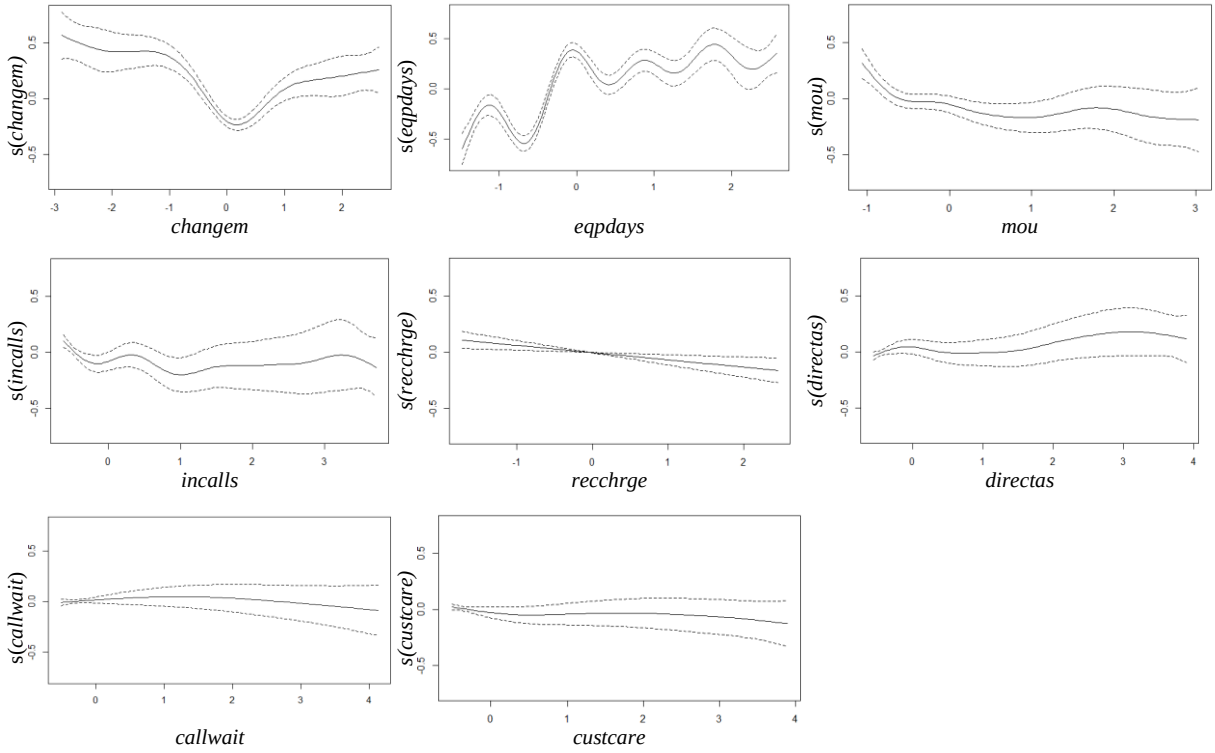


Figure 3: Visual representation of the eight penalized cubic regression spline terms in the Cell2cell SRE-SGL model

The following observations emerge from Table 4. First, the model contains 21 terms: 6 rules, 7 linear terms and 8 splines. Hence, the model illustrates well how SRE-SGL is capable of revealing linear effects, nonlinear effects as well as interaction effects. Second, investigating the nature of the impact of terms on customer churn is straightforward. The 4 most important terms are univariate: linear basis functions for *retcalls* and *revenue*, and *splines* for *changem* and *eqpdays*. Positive linear effects exist for *retcalls*, *revenue* and *changem_M*; negative ones for *creditde*, *recchrge* and *webcap* (in order of importance). Splines, visualized in Figure 3, reveal varying non-linear relationships to the probability to churn. Second, the rules reveal the existence of interaction effects. Closer inspection reveals that *eqpdays* interacts with two variables: *recchrge* and *retcalls*. Both are represented by three rules each, that are consistent in terms of coefficient sign and rule conditions.

5.2.2 SRE-SGL Comparison to Rule and Spline-Rule Ensembles

Next, we wish to compare the SRE-SGL model to a conventional RE and SRE model fit to the same data set (Tables A.1 and A.2 in Appendix A, respectively). This comparison illustrates the beneficial impact of SGL regularization with regards to interpretability on multiple accounts. This comparison involves three dimensions: (i) model structure and interpretation, (ii) variable importance and (iii) isolated variable effects.In terms of model structure and interpretation, substantial differences emerge that favor SRE-SGL in terms of interpretability. First, in terms of model size, the SGL led to a more compact model for SRE-SGL in comparison to rule and spline-rule ensemble models. Both benchmark models contain more terms: 31 and 38, respectively, versus 21 for the SRE-SGL model. Despite this large difference in model size, the predictive performance of SRE-SGL is better than RE and similar to SRE (as shown in Table 2). Second, the absence of structured regularization in conventional lasso regularization in rule and spline-rule ensembles compromises interpretability due to the presence of conflicting rules. For example, consider the interaction effect between *eqpdays* and *retcalls*. Both the rule and spline-rule ensemble model also recognize their interaction by selection rules defined on both. However, the nature of the interaction effect is much harder to disentangle due to (i) the number of rules that capture the interaction effect (7 for RE and SRE, versus 3 in SRE-SGL), and (ii) the presence of seemingly conflicting rules. For example, consider rules that depend on the variables *retcalls* and *eqpdays*. terms 7 and 25 in the rule ensemble model: (*retcalls* < 2.3026 * *eqpdays* < -0.2996; coefficient -0.0681) versus (*retcalls* ≥ 2.3026 * *eqpdays* < -0.2913; coefficient 0.0414). Such problems also emerge in the SRE model, and for the interaction between *recchrge* and *retcalls*. While in the SRE-SGL model, the sparse group lasso did not prevent a selection of multiple rules defined on both pairs of variables, within-group level shrinkage resulted in a more stringent selection of consistent rules. Third, the SRE model shows how in the absence of structured regularization, univariate terms can also cause conflicts and jeopardize model interpretability. For example: *changem* and *recchrge* occur both as a linear term and as a spline in the model. *Eqpdays* occurs as a spline, and in 8 univarate rules with varying coefficient signs. These problems are lifted in the SRE-SGL model. The exception is the double occurrence of *recchrge*, which shows that the introduction of within-group shrinkage in SGL is not absolute.

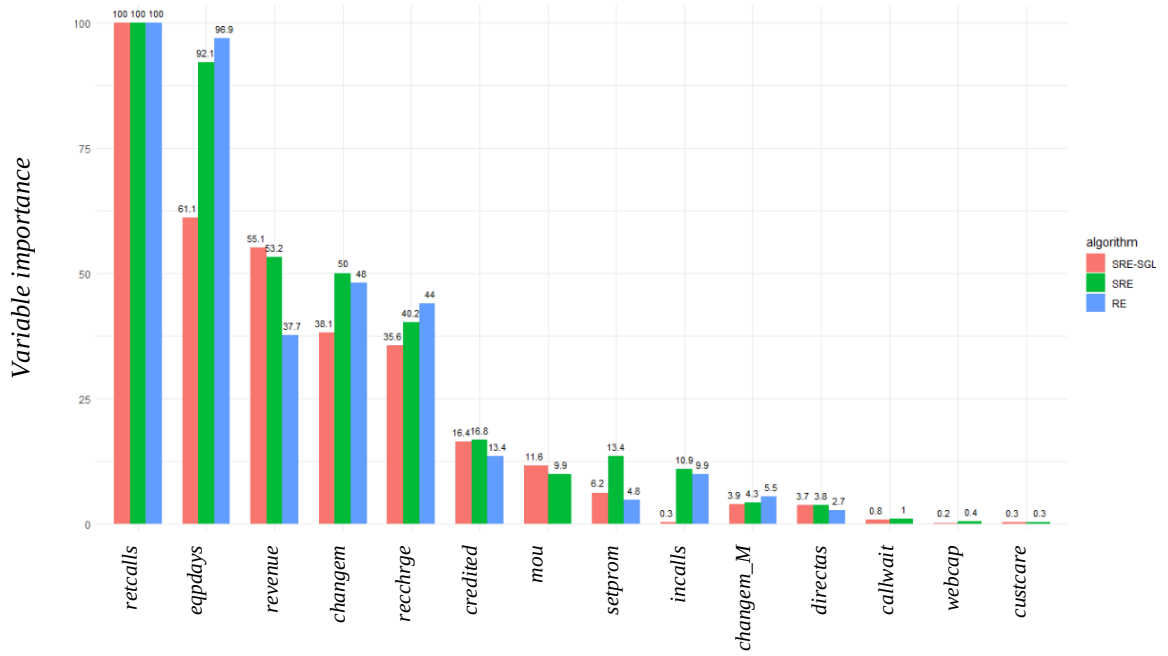


Figure 4: Cell2cell SRE-SGL, SRE and RE model variable importances

A second comparison analyses the importance of individual variables in the models. Figure 4 presents variable importance measures for all variables in the SRE-SGL, SRE and RE models, rescaled so that the most important variable receives a score of 100. These results show consistency between SRE-SGL, SRE and RE models, despite their differing model structures. In total, 14 variables appear in the SRE-SGL and SRE models, while there are 10 in the RE model. SRE-SGL, SRE and RE agree on the five most important variables (with an importance level above 20) are *retcalls*, *eqpdays*, *revenue*, *changem* and *recchrge*. These are the variables that dominate both as univariate terms, and in multiple multivariate rules in the model.

A third dimension on which SRE-SGL is compared to SRE and RE is the analysis of isolated variables effects. Our intention is to analyze whether the introduction of structured regularization in SRE-SGL substantially alters the nature of the effects found in regular SRE and RE models. To this end, the isolated variable effects are visualized in Figure 5 that shows the corresponding partial dependence plots for the variables in the model, ordered by importance (defined in Figure 4) for SRE-SGL, SRE and RE models.

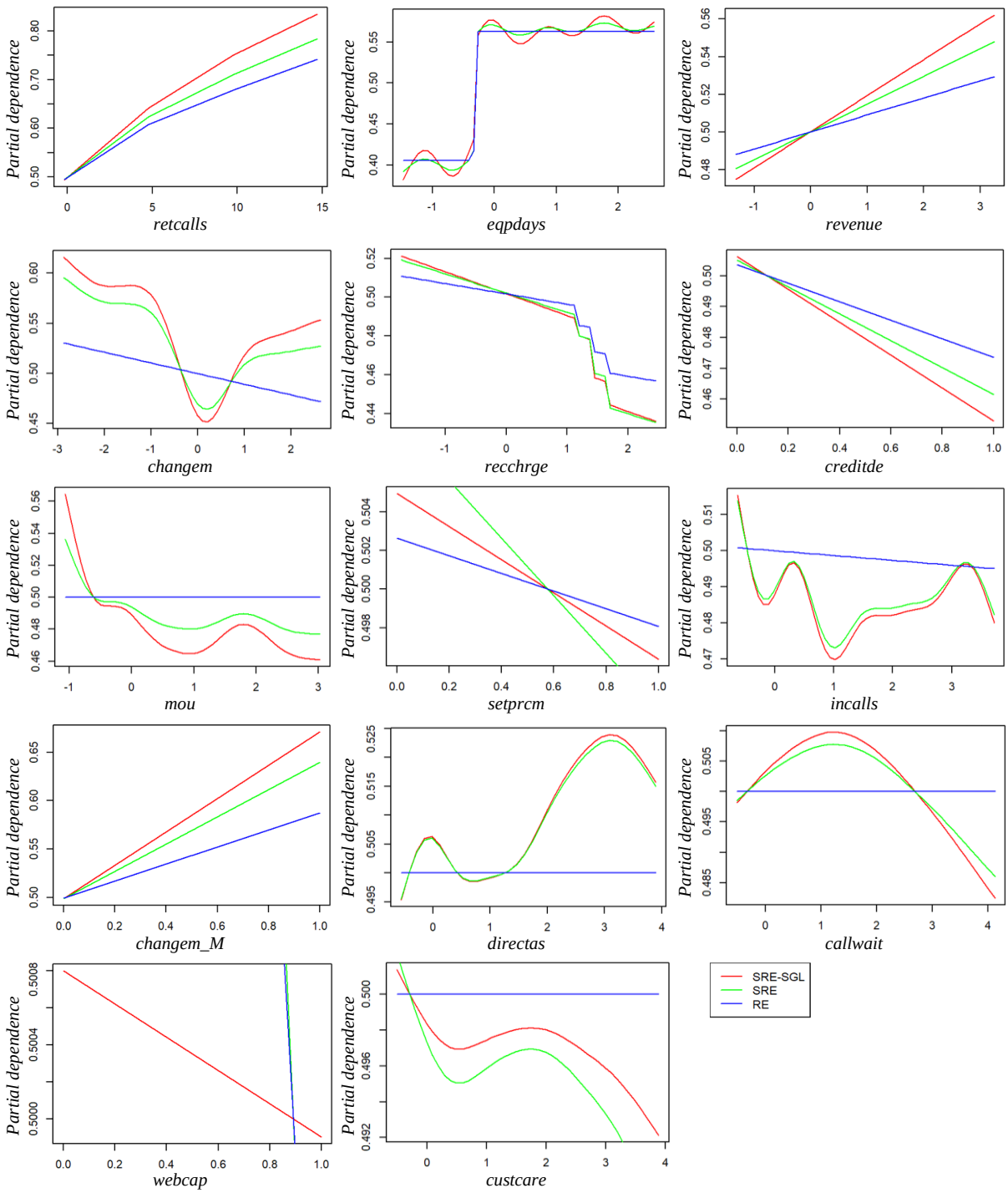


Figure 5: Partial dependence plots for selected variables: a comparison of the SRE-SGL, SRE and RE Cell2cell models

A comparison of these dependence plots leads to the following observations. First, effects in the SRE-SGL and SRE models are highly similar. This provides evidence that the altered model structure due to the introduction of structured regularization does in fact not substantially alter the isolated effects of individual variables. The enforcing of a grouping structure on model terms in SGL leads to a model

that is easier to understand, yet is similar in its functioning. Second, these plots highlight the added value of adding spline functions to rules and linear basis functions for churn prediction. Non-linear effects that revealed in SRE and SRE-SGL models are either not incorporated (e.g. *mou*, *directas*), or substituted by a linear effect (e.g. *changem*, *incalls*) or a piecewise linear effect (*eqpdays*) in the RE model. Note that such a piecewise linear function requires multiple rules to be simultaneously present in the model (as demonstrated by the many rules defined on *eqpdays* in the RE model in Table A.1). Third, it is useful to compare these partial dependence visualizations with the SRE-SGL model described earlier. The selected linear and spline terms can be easily recognized, the partial dependence function for the variable *eqpdays* is a composite of the spline term and the rules that feature the variable while the piecewise linear plots for *retcalls* and *recchrge* summarize the linear effects and rules in which they emerge.

An alternative use of partial dependence functions is the analysis of variable interactions. An analysis of the interaction effects in the SRE-SGL model is available in Appendix E.

6 Conclusions, limitations and directions for future research

Customer churn prediction is an important instrument in companies' retention management strategies. Such models ought to be as accurate as possible albeit not at the expense of decreased interpretability. Ensemble methods have been gaining critical acclaim since many years, mostly due to their association to strong predictive accuracy. However, in contexts where decision makers attribute high value to interpretability, their black-box nature compromises their potential deployment. RE and SRE constitute a family of classifiers tailored to reconcile these seemingly conflicting objectives. Based on rules extracted from decision trees, spline terms and simple linear terms, they offer increased flexibility over other established methods that are easy to understand, such as decision trees or logistic regression. Yet, this increased model complexity, that could be seen as the cost of this increased flexibility, is very limited thanks to regularization, i.e. the shrinkage of the full set of candidate model terms.

Since rules, splines and linear terms are essentially based on the same set of variables, interpretation of a rule- or a spline-rule ensemble model could become less straightforward when regularization

shrinkage does not prevent such terms from being selected simultaneously. To remedy this, we propose SRE-SGL, spline-rule ensembles with structure regularization through sparse group lasso regularization. We define a straightforward indexing function to group terms when they share the same set of defining variables. Through sparse group lasso regularization, term selection is driven by shrinkage at two levels: the between-group level and the within-group level. SRE-SGL aims for accurate models that consist of as few terms of as few variable groups as possible. Extensive experiments on a large set of churn prediction data sets confirm the highly competitive nature of SRE-SGL in terms of two dimensions it aims to reconcile: predictive accuracy and interpretability. Specifically, results demonstrate how SRE-SGL outperforms a decision tree, logistic regression, GAM, MARS and a random forest model on most datasets. It also consistently outperforms conventional RE models, and the introduction of structured regularization does not result in a disadvantage over standard SRE. Model interpretability of SRE-SGL was assessed in detail and compared to RE and SRE by means of a case study, investigating churn prediction in the setting of a telecommunications company. An analysis by means of variable importance measures and partial dependence functions demonstrates that the effects captured by SRE-SGL are highly similar to those found in the RE and SRE models. However, the SRE-SGL model is simpler in nature and its interpretability, unlike conventional RE and SRE models, is not compromised by inconsistent or conflicting model terms or term effects based on the same variables.

The contributions of this study are thus the following: (i) we introduce spline-rule ensembles to the field of customer churn prediction and demonstrate their ability to deliver insightful yet accurate models; (ii) we propose SRE-SGL as an extension to spline-rule ensembles that retains the qualities of spline-rule ensembles yet avoids the pitfall of reduced model interpretability due to conflicting model term selection, (iii) an extensive benchmark study is conducted to determine how SRE-SGL performs in comparison to well-established competing algorithms balancing accuracy and interpretability, and (iv) a case study is conducted to illustrate how SRE-SGL avoids the issues described above and achieves a higher degree of interpretability in comparison to conventional rule and spline-rule ensembles.

Note that a number of limitations of this study could be identified. First, our experimental comparison evaluated two well-known metrics for assessing classifier performance in the domain of

CCP with a widespread adoption and recognition in practice: top-decile lift and AUC. Recently, promising metrics that integrate cost and profit considerations in their evaluation gained popularity in the domain, such as the expected maximum profit criterion [64]. Future research should evaluate SRE-SGL on the basis of such metrics, and we intend to extend SRE-SGL so that these metrics guide decision tree training as well as regularization. Second, this study is exclusively focused on customer churn prediction which is one of the most established applications in the broader field of marketing analytics. Future research should explore the viability of SRE-SGL for other tasks. Third, regularized regression is one strategy for reducing a large set of terms and combining them. Other strategies that are common for the practice of ensemble selection of ensemble pruning such as optimization, could also be deployed for term selection as well as integrating variable group selectivity. Fourth, SRE-SGL, like RE and SRE, involves the initial creation of a large set of rules and terms and sometimes overly complex rules can still be retained in the final model. Follow-up research should explore strategies for the definition of preliminary term filters that enforce a priori selection. This could be a viable option to further constrain models in pursuit of enhanced model interpretability. Finally, data preprocessing practices such as feature selection, class imbalance and outlier treatment are known to impact results. While this study based its experimental set-up on prior literature, future research could revisit the impact of such practices on the novel algorithms presented in this study.

References

[1] A. Hagius, J. Wright, When Data Creates Competitive Advantage, Harv. Bus. Rev. (2020).

[2] A. De Caigny, K. Coussement, K.W. De Bock, A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees, Eur. J. Oper. Res. 269 (2018) 760–772.

[3] E. Ascarza, S.A. Neslin, O. Netzer, Z. Anderson, P.S. Fader, S. Gupta, B.G.S. Hardie, A. Lemmens, B. Libai, D. Neal, F. Provost, R. Schrift, In Pursuit of Enhanced Customer Retention Management: Review, Key Issues, and Future Directions, Cust. Needs Solut. (2018).

[4] W. Verbeke, K. Dejaeger, D. Martens, J. Hur, B. Baesens, New insights into churn prediction in the telecommunication sector: A profit driven data mining approach, Eur. J. Oper. Res. 218 (2012) 211–229.

[5] M. Óskarsdóttir, C. Bravo, W. Verbeke, C. Sarraute, B. Baesens, J. Vanthienen, Social network analytics for churn prediction in telco: Model building, evaluation and network architecture, Expert Syst. Appl. 85 (2017) 204–220.

[6] A. Gustafsson, M.D. Johnson, I. Roos, The Effects of Customer Satisfaction, Relationship Commitment Dimensions, and Triggers on Customer Retention, J. Mark. 69 (2005) 210–218.

[7] S.A. Neslin, S. Gupta, W. Kamakura, J.X. Lu, C.H. Mason, Defection detection: Measuring and understanding the predictive accuracy of customer churn models, J. Mark. Res. 43 (2006) 204–211.

[8] B. Masand, P. Datta, D.R. Mani, B. Li, CHAMP: A prototype for automated cellular churn prediction, Data Min. Knowl. Discov. 3 (1999) 219–225.

[9] K.W. De Bock, D. Van Den Poel, Reconciling performance and interpretability in customer churn prediction using ensemble learning based on generalized additive models, Expert Syst. Appl. 39 (2012).

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

[11] J.H. Friedman, B.E. Popescu, Predictive learning via rule ensembles, Ann. Appl. Stat. 2 (2008) 916–954.

[12] K.W. De Bock, The best of two worlds: Balancing model strength and comprehensibility in business failure prediction using spline-rule ensembles, Expert Syst. Appl. 90 (2017).

[13] F. Klepsch, G.F. Ecker, Impact of the Recent Mouse P-Glycoprotein Structure for Structure-Based Ligand Design, Mol. Inform. 29 (2010) 276–286.

[14] T. Ouyang, S. Ray, M. Allman, M. Rabinovich, A large-scale empirical analysis of email spam detection through network characteristics in a stand-alone enterprise, Comput. Networks. 59 (2014) 101–121.

[15] N. Simon, J. Friedman, T. Hastie, R. Tibshirani, A sparse-group lasso, J. Comput. Graph. Stat. 22 (2013) 231–245.

[16] M.T. Ribeiro, S. Singh, C. Guestrin, "Why should i trust you?" Explaining the predictions of any classifier, in: Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., 2016.

[17] D. Martens, F. Provost, Explaining data-driven document classifications, MIS Q. Manag. Inf. Syst. (2014).

[18] O. Biran, C. Cotton, Explanation and Justification in Machine Learning: A Survey, in: IJCAI-17 Work. Explain. AI, 2017.

[19] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, Artif. Intell. (2019).

[20] C. Molnar, Interpretable Machine Learning. A Guide for Making Black Box Models Explainable, 2019.

[21] K. Coussement, S. Lessmann, G. Verstraeten, A comparative analysis of data preparation algorithms for customer churn prediction: A case study in the telecommunication industry, Decis. Support Syst. (2016).

[22] A. De Caigny, K. Coussement, K.W. De Bock, S. Lessmann, Incorporating textual information in customer churn prediction models based on a convolutional neural network, Int. J. Forecast. (2019).

[23] S.A. Neslin, S. Gupta, W. Kamakura, J.X. Lu, C.H. Mason, Defection detection: Measuring and understanding the predictive accuracy of customer churn models, J. Mark. Res. 43 (2006) 204–211.

[24] K. Coussement, D. Van den Poel, Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques, Expert Syst. Appl. 34 (2008) 313–327.

[25] W. Verbeke, D. Martens, C. Mues, B. Baesens, Building comprehensible customer churn prediction models with advanced rule induction techniques, Expert Syst. Appl. 38 (2011) 2354–2364.

[26] M.A.H. Farquad, V. Ravi, S.B. Raju, Churn prediction using comprehensible support vector

machine: An analytical CRM application, Appl. Soft Comput. J. (2014).

[27] V.L. Miguéis, A. Camanho, J. Falcão e Cunha, Customer attrition in retailing: An application of Multivariate Adaptive Regression Splines, Expert Syst. Appl. 40 (2013) 6225–6232.

[28] J.H. Friedman, Multivariate Adaptive Regression Splines, Ann. Stat. 19 (1991) 1–67.

[29] K. Coussement, D.F. Benoit, D. Van den Poel, Improved marketing decision making in a customer churn prediction context using generalized additive models, Expert Syst. Appl. 37 (2010) 2132–2143.

[30] K. Coussement, K.W. De Bock, Customer churn prediction in the online gambling industry: The beneficial effect of ensemble learning, J. Bus. Res. 66 (2013) 1629–1636.

[31] K.W. De Bock, D. Van Den Poel, Reconciling performance and interpretability in customer churn prediction modeling using ensemble learning based on generalized additive models, Expert Syst. Appl. 39 (2012) 6816–6826.

[32] K.W. De Bock, K. Coussement, D. Van den Poel, Ensemble classification based on generalized additive models, Comput. Stat. Data Anal. 54 (2010).

[33] J. Qi, L. Zhang, Y. Liu, L. Li, Y. Zhou, Y. Shen, L. Liang, H. Li, ADTreesLogit model for customer churn prediction, Ann. Oper. Res. (2009).

[34] A. Backiel, B. Baesens, G. Claeskens, Predicting time-to-churn of prepaid mobile telephone customers using social network analysis, J. Oper. Res. Soc. (2016).

[35] T. Verbraken, W. Verbeke, B. Baesens, Profit optimizing customer churn prediction with Bayesian network classifiers, Intell. Data Anal. (2014).

[36] E. Lima, C. Mues, B. Baesens, Domain Knowledge Integration in Data Mining Using Decision Tables: Case Studies in Churn Prediction, J. Oper. Res. Soc. 60 (2009) 1096–1106.

[37] L.I. Kuncheva, Combining pattern classifiers: methods and algorithms, John Wiley & Sons, Hoboken, New Jersey, 2004.

[38] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: L. Saitta (Ed.), Thirteen. Int. Conf. Mach. Learn. (ICML 1996), Morgan Kauffman, Bari, Italy, 1996: pp. 148–156.

[39] L. Breiman, Bagging predictors, Mach. Learn. 24 (1996) 123–140.

[40] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.

[41] J.J. Rodríguez, L.I. Kuncheva, C.J. Alonso, I.C. Society, L.I. Kuncheva, C.J. Alonso, Rotation forest: A new classifier ensemble method, IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 1619–1630.

[42] W.W. Cohen, Y. Singer, A Simple, Fast, and Effective Rule Learner, in: Proc. Sixt. Natl. Conf. Artif. Intell. Elev. Innov. Appl. Artif. Intell. Conf. Innov. Appl. Artif. Intell., American Association for Artificial Intelligence, USA, 1999: pp. 335–342.

[43] J.H. Frieman, B.E. Popescu, Predictive learning via rule ensembles, Ann. Appl. Stat. 2 (2008) 916–954.

[44] J. Błaszczyński, K. Dembczyński, W. Kotłowski, R. Słowiński, M. Szelkag, Ensemble of decision rules, Found. Comput. Decis. Sci. 31 (2006) 221–232.

[45] K. Dembczyński, W. Kotłowski, R. Słowiński, Maximum Likelihood Rule Ensembles, in: Proc. 25th Int. Conf. Mach. Learn., Association for Computing Machinery, New York, NY, USA, 2008: pp. 224–231.

[46] W. Yang, S. Zhang, Y. Chen, Y. Chen, W. Li, H. Lu, Mining diagnostic rules of breast tumor on ultrasound image using cost-sensitive RuleFit method, in: 2008 3rd Int. Conf. Intell. Syst. Knowl. Eng., 2008: pp. 354–359.

[47] D. Mohapatra, B. Subudhi, Weighted majority rule ensemble classifier for sensor fault

classification for plasma position control in Tokamaks, Fusion Eng. Des. 160 (2020) 111969.

[48] J.M. J. Debrulle, P. Steffens, K. W. De Bock, S. De Winne, Configurations of Business Founder Resources, Strategy and Environment Determining New Venture Performance, J. Small Bus. Manag. (2021).

[49] T. Shimokawa, L. Li, K. Yan, S. Kitamura, M. Goto, Modified Rule Ensemble Method for Binary Data and Its Applications, Behaviormetrika. 41 (2014) 225–244.

[50] R. Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc. Ser. B. 58 (1996) 267–288.

[51] S.N. Wood, Generalized additive models: an introduction with R, CRC press, 2017.

[52] M. Yuan, Y. Lin, Model selection and estimation in regression with grouped variables, J. R. Stat. Soc. Ser. B (Statistical Methodol. 68 (2006) 49–67.

[53] L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and regression trees, Wadsworth International Group, Belmont, CA, 1984.

[54] T. Hastie, R. Tibshirani, Generalized Additive Models, Chapman and Hall, London, 1990.

[55] P. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Addison Wesley, Boston, MA, 2006.

[56] J. Burez, D. Van den Poel, Handling class imbalance in customer churn prediction, Expert Syst. Appl. 36 (2009) 4626–4636.

[57] P. Craven, G. Wahba, Smoothing noisy data with spline functions, Numer. Math. 31 (1978) 377–403.

[58] S.N. Wood, Stable and efficient multiple smoothing parameter estimation for generalized additive models, J. Am. Stat. Assoc. 99 (2004) 673–686.

[59] J. Demšar, Statistical Comparisons of Classifiers over Multiple Data Sets, J. Mach. Learn. Res. 7 (2006) 1–30.

[60] S. García, A. Fernández, J. Luengo, F. Herrera, Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power, Inf. Sci. (Ny). 180 (2010) 2044–2064.

[61] J. Hastie, Trevor, Tibshirani, Robert, Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Second Edition), 2009.

[62] M. Bogaert, M. Ballings, D. Van den Poel, The added value of Facebook friends data in event attendance prediction, Decis. Support Syst. 82 (2016) 26–34.

[63] J. Demšar, Statistical Comparisons of Classifiers over Multiple Data Sets, J. Mach. Learn. Res. 7 (2006) 1–30.

[64] T. Verbraken, W. Verbeke, B. Baesens, A novel profit maximizing metric for measuring classification performance of customer churn prediction models, IEEE Trans. Knowl. Data Eng. 25 (2013) 961–973.

Appendices

Appendix A: Cell2cell Rule Ensemble and Spline-Rule Ensemble Models

Term	Type	Term or rule specification	Coefficient	Rule support	Term importance
1	Linear term	*retcalls*	0.1666	-	100
2	Linear term	*changem*	-0.1095	-	61.9698
3	Linear term	*revenue*	0.0920	-	51.4809
4	Linear term	*recchrge*	-0.0544	-	30.5365
5	Rule	*recchrge* < 1.3767 *eqpdays* ≥ -0.2954	0.0726	0.5529	20.4608
6	Rule	*recchrge* < 1.1742 *eqpdays* ≥ -0.2954	0.0679	0.5483	19.1421
7	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2996	-0.0681	0.3901	18.8075
8	Linear term	*creditde*	-0.1001	-	18.3641
9	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2954	-0.0664	0.3918	18.3578
10	Rule	*recchrge* < 1.6328 *eqpdays* ≥ -0.2954	0.0652	0.5640	18.3211
11	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2913	-0.0552	0.3947	15.2900
12	Linear term	*webcap*	-0.0765	-	13.5115
13	Rule	*retcalls* < 2.3026 *eqpdays* < -0.3615	-0.0386	0.3654	10.5356
14	Rule	*eqpdays* ≥ -0.2996	0.0351	0.5920	9.7650
15	Rule	*eqpdays* < -0.2996	-0.0349	0.4080	9.7185
16	Rule	*eqpdays* ≥ -0.2954	0.0339	0.5903	9.4440
17	Rule	*eqpdays* < -0.2954	-0.0339	0.4097	9.4374
18	Linear term	*incalls*	-0.0135	-	7.5088
19	Rule	*eqpdays* < -0.3161	-0.0238	0.4017	6.6227
20	Linear term	*setprcm*	-0.0234	-	6.5475
21	Rule	*eqpdays* ≥ -0.3160	0.0233	0.5983	6.4581
22	Rule	*eqpdays* ≥ -0.2913	0.0215	0.5873	5.9955
23	Rule	*eqpdays* < -0.2913	-0.0214	0.4127	5.9731
24	Linear term	*changem_M*	0.0770	-	3.6709
25	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.2913	0.0414	0.0180	3.1160
26	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.3615	0.0415	0.0173	3.0616
27	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.296	0.0288	0.0179	2.1605
28	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.2954	0.0280	0.0179	2.1015
29	Rule	*eqpdays* < -0.3615	-0.0049	0.3827	1.3619
30	Rule	*eqpdays* ≥ -0.3615	0.0049	0.6173	1.3523
31	Rule	*recchrge* ≥ 1.3767 *eqpdays* ≥ -0.2954	-0.0113	0.0374	1.2163

Table A.1: The Cell2cell rule ensemble model: terms, term types, rule conditions, rule support and term importance. Terms are sorted by their importance.

Term	Type	Term or rule specification	Coefficient	Rule support	Term importance
1	Linear term	*retcalls*	0.2107	-	100
2	Linear term	*revenue*	0.1523	-	67.4154
3	Spline	s(*changem*)	0.3692	-	39.0854
4	Linear term	*recchrge*	-0.0505	-	22.4276
5	Linear term	*creditde*	-0.1467	-	21.2720
6	Linear term	*changem*	-0.0438	-	19.6160
7	Spline	s(*eqpdays*)	0.1242	-	18.3232
8	Rule	*recchrge* < 1.3767 *eqpdays* ≥ -0.29547	0.0785	0.5529	17.4713
9	Linear term	*setprcm*	-0.0769	-	17.0326
10	Rule	*recchrge* < 1.6328 *eqpdays* ≥ -0.2954	0.0705	0.5640	15.6602
11	Rule	*recchrge* < 1.1742 *eqpdays* ≥ -0.2954	0.0693	0.5483	15.4544
12	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2996	-0.0697	0.3901	15.2346
13	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2954	-0.0665	0.3918	14.5401
14	Linear term	*webcap*	-0.0987	-	13.7891
15	Spline	s(*mou*)	0.1646	-	10.1537
16	Rule	*retcalls* < 2.3026 *eqpdays* < -0.2913	-0.0462	0.3947	10.1205
17	Rule	*eqpdays* ≥ -0.2996	0.0369	0.5920	8.1317
18	Rule	*eqpdays* < -0.2996	-0.0366	0.4080	8.0486
19	Rule	*eqpdays* ≥ -0.2954	0.0343	0.5903	7.5649
20	Rule	*eqpdays* < -0.2954	-0.0342	0.4097	7.5306
21	Spline	s(*incalls*)	0.1299	-	5.4361
22	Rule	*retcalls* < 2.3026 *eqpdays* < -0.3614	-0.0226	0.3654	4.8747
23	Linear term	*changem_M*	0.1264	-	4.7648
24	Rule	*eqpdays* < -0.3160	-0.0166	0.4017	3.6533
25	Rule	*eqpdays* ≥ -0.3161	0.0160	0.5983	3.5121
26	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.3615	0.0562	0.0173	3.2754
27	Rule	*recchrge* ≥ 1.3767 *eqpdays* ≥ -0.2954	-0.0369	0.0374	3.1351
28	Rule	*eqpdays* < -0.2913	-0.0142	0.4127	3.1247
29	Rule	*eqpdays* ≥ -0.2913	0.0136	0.5873	2.9880
30	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.2913	0.0469	0.0180	2.7890
31	Rule	*recchrge* ≥ 1.6328 *eqpdays* ≥ -0.2954	-0.0348	0.0264	2.4933
32	Linear term	*mou*	-0.0054	-	2.3950
33	Spline	s(*recchrge*)	0.0520	-	1.4844
34	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.2996	0.0248	0.0179	1.4729
35	Rule	*retcalls* ≥ 2.3026 *eqpdays* < -0.2954	0.0231	0.0179	1.3708
36	Spline	s(*directas*)	0.0605	-	1.2405
37	Spline	s(*custcare*)	0.0354	-	0.5295
38	Spline	s(*callwait*)	0.0373	-	0.3732

Table A.2: The Cell2cell spline-rule ensemble model: terms, term types, rule conditions, coefficients, rule support and term importance. Terms are sorted by their importance.

Appendices B - E

Appendices B-E are available as online supplementary materials available for download at https://github.com/koendebock/SRE-SGL/blob/master/SRE-SGL_supplementary_materials.pdf.