EXPLANATORY VARIABLE SELECTION WITH BALANCED CLUSTERING IN CUSTOMER CHURN PREDICTION

[a] MARTIN FRIDRICH

Department of Informatics, Faculty of Business and Management, Brno University of Technology, Kolejní 2906/4, 612 00 Brno, Czech Republic
email: [a] fridrichmartin@yahoo.com

Abstract: The interest in customer relationship management has been fueled by the broad adoption of customer-centric paradigm, rapid growth in data collection, and technology advances for more than the past 15 years. It becomes hard to identify and interpret meaningful patterns in customer behavior; thus the goal of the paper is to compare multiple explanatory variable selection procedures and their effect on a customer churn prediction model. Filter and wrapper concepts of variable selection are examined, moreover, the runtime of the machine learning pipeline is improved by the novel idea of balanced clustering. Classification learners are incorporated with regard to simplicity and interpretability (LOGIT, CIT) and complexity and proven performance on a given dataset (RF, RBF-SVM). In addition, we show that when combined with learner capable of embedded feature selection, explicit variable selection scheme does not necessarily lead to performance improvement. On the other hand, RBF-SVM learner with no such ability benefits from relevant selection procedure in all expected aspects, including classification performance and runtime, problem comprehensibility, data storage.

Keywords: customer churn prediction, customer relationship management, feature selection, machine learning, variable importance

1 Introduction

Customer relationship management (CRM) became a topic of interest with a shift to the customer-centric paradigm. It aims to create, retain, and strengthen a relationship with customers while maintaining profits and revenue. Over the past 15 years, CRM has been augmented by progress in data collection and technology, enabling to tackle its challenges with a new set of tools, i.e., machine learning. An important objective of customer relationship management is to minimize customer churn, where term customer churn refers to affinity to cease business with the company in a given time. Churn reduction is usually motivated by a difference between underlying unitary costs of customer acquisition and customer retention, even though there are more benefits to it (Gronwald, 2017; Gupta et al., 2004; Torkzadeh et al., 2006).

To retain customers, prediction models are required to identify early churn signals and flag customers at high risk of leaving. In an environment, with rapid growth in data generation and collection, it becomes increasingly challenging to detect meaningful patterns and extract useful knowledge. Hence the aim of the paper is to examine the explanatory variable selection procedure and its effect on the performance of the churn prediction model. It is generally assumed that the explanatory variable selection procedure improves learner prediction performance, ability to generalize the problem, comprehensibility, reduce computational runtime and reduce storage requirements (acc. Aggarwal, 2014; Bagherzadeh-Khiabani et al., 2016).

2 Explanatory variable selection

The merit of explanatory variable selection is to find a subset of explanatory variables, which highly discriminates response variable. One can distinguish three procedure types – filter, wrapper and others (embedded, hybrid) however opinions on the matter might differ (Aggarwal, 2014; Bolón-Canedo et al., 2013; Bagherzadeh-Khiabani et al., 2016; Duda et al., 2012). We focus solely on filter and wrapper selection procedures. The task of dimensionality reduction is also tackled with feature extraction methods (PCA, LDA, CCA, Isomap, Autoencoder, etc.) since they project original features into new feature space while losing original comprehensibility (Aggarwal, 2014), they are not included.

Filter selection – FS relies on data properties without utilizing any classification learner. The procedure consists of two steps, (1) features are ranked according to the chosen criterion, (2) highly ranked features are selected. Univariate filters account only for a feature-class relationship; however, multivariate filters explore feature set-class relationship; hence, the former is inferior to the latter in handling redundant features.

Wrapper selection – As opposed to FS, WS adopts classification learner to estimate the quality of the feature set. Considering specific classification learner wrapper selection consists of three steps, (1) searching subset of features (2) evaluating the selected subset of features by the learner (3) repeating (1) and (2) until a stopping criterion is met. WS outperforms FS in terms of prediction quality of final learner, although the procedure can be computationally very expensive.

Others – In addition to FS and WS procedures, scientific literature depicts two more categories of selection methods, (1) embedded procedures – feature selection is included in the phase of learner fitting (i.e., logistic regression with L1 regularization, tree-based methods), which might reduce computational time (2) hybrid procedures – usually sequential combination of FS and WS method.

The explanatory variable selection domain broadly intersects with fields of machine learning (see Aggarwal, 2014; Arauzo-Azofra et al., 2008; Bolón-Canedo et al., 2013; Dash, Liu, 2003; Duda et al., 2012; Hall, 1999; Kononenko, 1994; Shakil Pervez, Farid, 2015), biostatistics and high-throughput biology (see Bagherzadeh-Khiabani et al., 2016; Guyon et al., 2002; Gilhan et al., 2010; Zhu et al., 2010; Chu et al., 2011). In customer churn domain, applications are limited and default to an evaluation of only a few feature selection/extraction methods (see Verbeke et al., 2012; Xiao et al., 2015; Spanoudes, Nguyen, 2017; Subramanya, Somani, 2017; Vijaya, Sivasankar, 2018). Hence, our goal is to examine the performance of multiple approaches to explanatory variable selection and to compare the results with literature utilizing the same customer churn dataset.

2.1 Filter selection

Fisher score – FS is univariate selection method, returns feature ranks. Important features are expected to exhibit similar observed values in the one class and different observed values across different classes. This intuition is denoted in formula (1), where S_i stands for Fisher score, μ_{ij} and ρ_{ij}^2 are the mean and variance of i-th feature in the j-th class respectively n_j is the number of instances in the j-th class, and μ_i is the mean of the i-th feature (acc. Aggarwal, 2014; Bagherzadeh-Khiabani et al., 2016).

$$FS_i = \frac{\sum_{k=1}^{K} n_j (\mu_{ij} - \mu_i)^2}{\sum_{k=1}^{K} n_j \rho_{ij}^2} \qquad 1)$$

Entropy-based measures – EBMs are based on an idea of measuring uncertainty, the unpredictability of the variable. In the paper, three types of information measures are examined: (1) information gain, (2) information gain ratio, and (3) symmetrical uncertainty criterion.

Information gain (IG) is denoted in formula (2), where $H(f_i)$ represents entropy of i-th feature, $H(C)$ stands for class entropy, and $H(f_i|C)$ amounts to joint entropy of f_i and C. Features with high IG are considered necessary, this predicament also holds for IGR and SU (acc. Aggarwal, 2014; Bagherzadeh-Khiabani et al., 2016; Duda et al., 2012).

$$IG(f_i, C) = H(f_i) + H(C) - H(f_i|C) \qquad (2)$$

IG suffers from a bias towards multi-valued features, to correct that different metric was proposed – information gain ratio (IGR). IGR is denoted in formula (3) (acc. Aggarwal, 2014; Bagherzadeh-Khiabani et al., 2016; Duda et al., 2012).

$$IGR(f_i, C) = \frac{IG(f_i, C)}{H(f_i)} \quad (3)$$

IGR is limited by its asymmetry. To deal with both lack of symmetry and bias towards multi-valued features, symmetrical uncertainty criterion (SU) was suggested. SU is denoted in formula (4) (acc. Aggarwal, 2014; Bagherzadeh-Khiabani et al., 2016; Duda et al., 2012).

$$SU(f_i, C) = 2\frac{IG(f_i, C)}{H(f_i) + H(C)} \quad (4)$$

OneR – OneR is univariate selection method, returns feature ranks. It creates a root-level decision tree for each feature and target class. For each such a tree, the error rate is calculated. Features with a low error rate are considered important (acc. Bagherzadeh-Khiabani et al., 2016).

Relief – Relief is multivariate selection method, returns feature ranks. It randomly samples observations and locates its nearest neighbor in the same and different target class; feature importance is adjusted subsequently. A significant feature set is assumed to have homogeneous values for each class, heterogeneous values across classes (Kononenko, 1994).

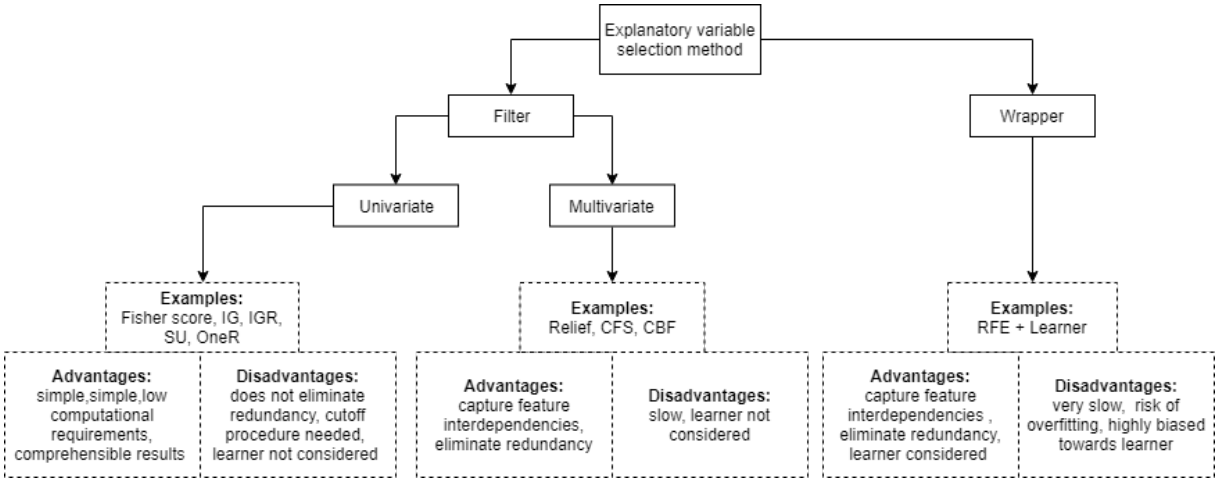


Figure 1. Categorization of explanatory variable selection methods, Source: 3, 4

Correlation-based feature selection – CFS is multivariate selection method, returns feature subset. CFS measures how are features in feature set correlated with each other and with the target class. A feature set with high correlation with class and low correlation amongst features is preferred. This intuition is denoted in formula (5), where M_S stands for heuristic "merit" of a feature subset S consisting of k features, $\overline{r_{cf}}$ is the mean feature-class correlation for $f \epsilon S$, and $\overline{r_{ff}}$ is the average feature-feature inter-correlation (Dash, Liu, 2003; Hall, 1999). Search through feature subset space is done through the best-first forward search.

$$M_S = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}} \quad (5)$$

Consistency-based filter – CBF is multivariate selection method, returns feature subset. CBF evaluates how consistently belong observations with the same set of feature values to target class (continuous feature values must be discretized). The algorithm finds a feature subset relying on Liu's consistency measure. Consistency measure is denoted in formula (6) (acc. Arauzo-Azofra, 2008). Search through feature subset space is also maintained with the best-first forward search.

$$Consistency = 1 - \frac{number\ of\ inconsistent\ observations}{number\ of\ observations} \quad (6)$$

2.2 Wrapper selection

Recursive feature elimination – RFE is popular multivariate selection method, returns feature ranks. The algorithm fits classification learner to the full feature set. Each feature is ranked using the classification learner (its coefficients / importance). At each iteration of the algorithm, top-ranked features are retained (low ranked features are eliminated), the classification learner is refit and scored. The feature set with learner's best performance is chosen. RFE was originally proposed with linear SVM (see Guyon et al., 2002), procedure, however, can be utilized with different classification learners. We combine RFE with classification methods LOGIT, RF, RBF-SVM. The author considers RFE to be wrapper selection procedure based on the implementation used (see Kuhn, 2008); however, opinions on the matter differ (see Aggarwal, 2014; Guyon et al., 2002).

3 Classification methods

From a machine learning viewpoint, customer churn prediction is perceived as a binary classification problem with a purpose to assign observations (customers) into one of two classes (churners, non-churners). There is a vast amount of research dedicated to classification method selection; however, we have decided to apply (1) simple and interpretable classification methods (LOGIT, CIT) and (2) more complex classification methods, with the proven performance considering given dataset (RF, RBF-SVM), acc. Verbeke et al. (2012).

Logistic regression – LOGIT is a parametric statistical method which estimates the probability of an event (discrete response variable), based on known circumstances (explanatory variables). LOGIT models tend to suffer from the influence of confounding factors and overfitting, to prevent that we used LOGIT with L1 and L2 regularization forms (Fan et al., 2008). LOGIT is straightforward to understand and interpret; it is also broadly used as classification baseline.

Conditional inference tree – CIT is non-parametric decision tree method (DT). Common implementations of DT tend to overfit and endure bias towards selected features. To address that, Hothorn et al. (2006) propose to base the splitting criterion on resampling and multiple inference tests, resulting in CIT. Its prediction ability is proven to be on par with pruned DT with no bias towards selected explanatory variables (see Horton et al., 2006; Horton, Zeileis, 2015).

Random forest – RF is non-parametric ensemble method which combines DTs such that each model is built upon randomly sampled explanatory variables (with replacement), votes of individual DTs are aggregated to form the prediction (Breiman, 2001). RF models are prone to overfitting and often produce satisfying prediction results without extensive hyperparameter search.

Support vector machine – Gaussian radial basis function SVM (RBF-SVM) is a non-parametric method that constructs hyperplane in high-dimensional space which has the largest distance (maximum-margin) between borderline observations (support vectors) while separating classes. Use of RBF kernel trick enables more complex boundaries in original feature space, which may lead to overfitting when not having enough observations (acc. Jin, Wang, 2012).

4 Research methodology

4.1 Dataset

We utilize public telecommunication dataset, originally published on UCI Machine learning repository, which is now part of the C50 package in CRAN. The dataset is popular in customer churn prediction research (see Verbeke et al., 2012; Vafeiadis et al., 2015; Mehreen et al., 2017) enabling broader discussion of results. It consists of 5000 observations, 19 explanatory variables (features), and 1 response variable (churn). The features are largely based on transactional data. Observed churn rate is 14.14 %.

4.2 Performance metrics

Accuracy – Performance of classification methods is routinely evaluated with confusion matrix and related measures. One of the popular metrics is ACC. It is defined as follows (Powers, 2011):

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \quad 7)$$

wherein numerator depicts a number of correctly classified positive (TP) and negative examples (TN), in the denominator we have sum a of correctly ($TP + TN$) and incorrectly classified examples ($FP + FN$). Accuracy is used for clear interpretability; however, it is threshold dependent and is not reliable when dealing with imbalanced classes.

Table 1. Churn dataset - variable names and data types

Variable name	Description	R dtype
state		factor
account_length	number of months as an active user	int
area_code		factor
international_plan	has an international plan (yes/no)	factor
voice_mail_plan	has voicemail plan (yes/no)	factor
number_vmail_messages	number of voice mail messages	int
total_day_minutes	total sum of day call minutes	num
total_day_calls	total number of day calls	int
total_day_charge	total sum of day charge	num
total_eve_minutes	total sum of evening call minutes	num
total_eve_calls	total number of evening calls	int
total_eve_charge	total sum of evening charge	num
total_night_minutes	total sum of night call minutes	num
total_night_calls	total number of night calls	int
total_night_charge	total sum of night charge	num
total_intl_minutes	total sum of international call minutes	num
total_intl_calls	total number of international calls	int
total_intl_charge	total sum of international charge	num
number_customer_service_calls	number of calls to customer service	int
churn	response variable	logi

Source: author

Top-decile lift – In retention campaign, the only a fraction of customers can be contacted and offered discount or premium service. To address that, TDL as an extension of ACC measure is often applied. It is calculated as a ratio of $ACCs$, with ACC for customers in top-decile propensity to churn (churn score) in the numerator and ACC for whole customer base in the denominator. TDL is popular for its practical implications; however, it is threshold dependent and ignores variations in fraction selection (Verbeke et al., 2012).

Area under the receiver operating curve – Classification model is expected to produce churn score $s = s(x)$, which is a function of feature vector x; probability density function of corresponding scores is described as $f_k(s)$, with cumulative distribution function $F_k(s)$ and two classes $k \in \{0,1\}$. AUC/ROC is then outlined in Eq. 8 (Hand, 2009).

$$AUC/ROC = \int_{-\infty}^{\infty} F_0(s) f_1(s) ds \quad 8)$$

AUC/ROC notion can be interpreted as a probability that randomly drawn member of class 0 will produce a lower churn score than randomly drawn member of class 1. AUC/ROC is the most popular measure of classification performance due to threshold independence (acc. Bradley, 1997), albeit it suffers from several conceptual issues (see Hand, 2009).

4.3 Experimental design and implementation

Performance of different feature selection techniques is examined through machine learning pipeline consisting of four main steps – (1) data processing, (2) feature selection, (3) model training and (4) model evaluation; their linkage is characterized in Fig. 2. To ensure the stability of the outcomes, the process is repeated 50 times. The pipeline is implemented in the R language for statistical programming, specifically in Microsoft R 3.5.1.

Data processing – Original churn dataset is randomly stratified into the train (60 % of examples) and test set (40 % of examples). Data transformations are performed on the train set and projected to the test set to prevent data leak. Non-binary factor columns are concealed with the one-hot encoding scheme. Numerical/integer features are expanded to 2nd-degree interaction terms, which results in a total of 158 explanatory variables. Consequently, all numerical/integer features are centered and scaled. Features with near zero variability are removed.

Feature selection – Processed train set serves as the only input to feature selection block. To address computational complexity and class imbalance in the feature selection procedure, we propose a balanced clustering method to reduce the number of observations. The algorithm is described with pseudo-code in Fig. 3. It is worth noting that the upper boundary for the expected number of examples per class is limited by properties of the train set. The procedure is implemented with clustering

around medoids and balancing classes with 250 observations per each, producing a total of 500 observations. The resulting set is then utilized over selection schemes.

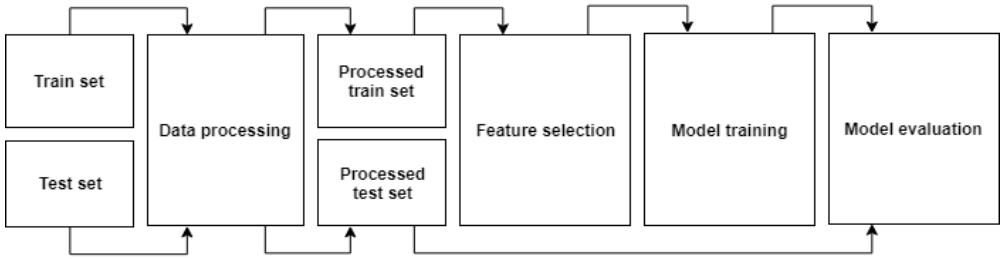


Figure 2. Conceptual depiction of machine learning pipeline, Source: author

To ensure the algorithm captures the original data structure, projection to two-dimensional space is made with Isomap embeddings in Fig. 4. The structure of the train set after balanced clustering resembles the original train set, the majority class of non-churners is largely under-sampled, which is to be expected.

We focus on two types of selection procedure – filter selection and wrapper selection. In filter-based selection whole train set is utilized at once, feature importance is estimated, unimportant features are filtered out. For wrapper-based selection, stratified 4-fold cross validation with 2 repeats is utilized in RFE procedure. RFE classification learner is subject to randomized hyperparameter search with 5 steps; target metric for both RFE selection and parameter search is set to AUC/ROC, as it is not subjective-dependent. Number of features to be selected is the function of each procedure, albeit univariate filter selection is set to return at least 20 explaining variables.

Algorithm 1:

1.1 for each class in target class do:
- 1.2 get feature data, where target class equals class
- 1.3 cluster features, set number of clusters to expected number of observations per target class
- 1.4 get the observation which is the nearest to each cluster center
- 1.5 add a class label to the selected observations
- 1.6 return the temporary results

1.7 row-bind temporary results to the feature train set

Figure 3. Balanced clustering for feature selection, Source: author

Model training and evaluation – Model training block digests processed train set and annotations of feature selection and classification method. A classifier is trained for all combinations of feature selection method and classification method. Experimental setup for randomized hyperparameter search is based on stratified 4-fold cross validation with 2 repeats; parameter search is done in 15 steps; its target metric is set to AUC/ROC. Final classification learners are built on top of the processed train set, feature selection, and randomized hyper-parameter search. Learner's performance on unseen data is estimated on the processed test set; to address bias-variance trade-off performance on the processed train set is also evaluated. Applied metrics are described in detail in the previous section.

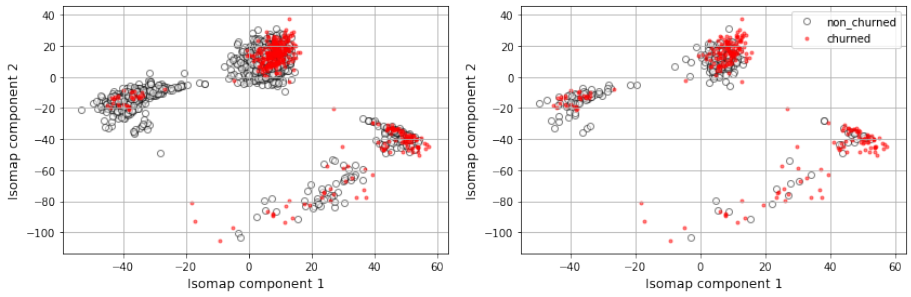


Figure 4. Structure of the original train set (left) and train set after balanced clustering (right), Source: author

5 Results

In order to summarize the performance of multiple approaches to explanatory variable procedures, mean point estimates across feature selection methods are depicted in Tab.3; the best indicators are marked in bold; 95 % confidence intervals for underlying distributions are constructed.

Classifiers combined with RFE selection show marginally better performance on both train and on test sets. Consistency, EBM schemes, and OneR display noteworthy behavior when significantly reducing the number of original features while (1) being almost on par with RFE procedures in all performance measures and (2) being less computationally demanding. Other selection procedures do not perform that well, which is induced by a considerable drop in retained features.

Statistical significance of a difference is assessed by paired t-tests with Bonferroni correction. Test performance of each selection scheme is compared to test performance without selection procedure; observations are paired on classification learners and pipeline repeats. H_0 states that true difference in sample means is equal to 0, H_A means the true difference in sample means is not equal to 0. We reject H_0 for all feature selection schemes except SVM-RFE, LR-RFE, RF-RFE on unadjusted $\alpha = 0.01$; this holds for all performance indicators. In other words, there is not enough evidence that SVM-RFE, LR-RFE, RF-RFE selection schemes improve test set

performance; on the other hand, previously mentioned procedures allow us to reduce explanatory variables by ~ 40 % while retaining the same level of classification performance as with original dataset. Other feature selection methods appear to lead to inferior results.

Table 2. Classification methods with respective parameters

Classification method	Optimized parameters	Implementation
LOGIT	regularization forms: {L1, L2 dual, L2 primal}, cost	Fan et al., 2008
CIT	max tree depth, p-value threshold	Hothorn, Zeileis, 2015
RF	number of selected predictors, splitting rule, minimal node size	Wright, Ziegler, 2017
SVM	kernel: {RBF}, cost, sigma	Karatzoglou et al., 2004

Source: author

To inspect explanatory variable importance in original feature space (Tab. 1), co-occurrence matrix of selection scheme-feature is constructed; the number of feature occurrence for both individual and interaction terms are included. Moreover, the co-occurrence matrix is scaled by the maximum possible incidence of a feature (scheme-feature pair for the procedure without feature selection). The result of the outlined steps is depicted using heatmap and dendrograms in Fig. 5; the explanatory variable state is not present as it is eliminated in data preprocessing step due to near-zero variance.

Table 3. Classification performance indicators aggregated by the feature selection method

feature selection method	number of features	feature selection runtime [s]	Train ACC (95 % CI)	Train AUC (95 % CI)	Train TDL (95 % CI)	test ACC (95 % CI)	test AUC (95 % CI)	test TDL (95 % CI)
CFS	10.3	6.1	0.923 (0.852, 0.994)	0.911 (0.801, 1.022)	5.678 (3.441, 7.914)	0.905 (0.860, 0.950)	0.874 (0.813, 0.935)	5.171 (3.462, 6.880)
Consistency	18.2	139.6	0.939 (0.870, 1.007)	0.929 (0.836, 1.021)	6.165 (4.432, 7.898)	0.919 (0.877, 0.960)	0.890 (0.848, 0.931)	5.696 (4.381, 7.011)
FS	24.8	0.1	0.920 (0.860, 0.979)	0.907 (0.797, 1.017)	5.632 (3.822, 7.442)	0.904 (0.867, 0.941)	0.868 (0.786, 0.950)	5.123 (3.809, 6.437)
Relief	25.4	179.9	0.915 (0.850, 0.980)	0.896 (0.759, 1.032)	5.444 (3.309, 7.579)	0.898 (0.860, 0.935)	0.852 (0.765, 0.940)	4.868 (3.430, 6.305)
IGR	44.4	0.6	0.937 (0.872, 1.002)	0.927 (0.831, 1.022)	6.124 (4.429, 7.819)	0.918 (0.878, 0.958)	0.889 (0.836, 0.941)	5.652 (4.281, 7.024)
IG	45.0	0.7	0.939 (0.876, 1.003)	0.930 (0.844, 1.016)	6.196 (4.603, 7.789)	0.920 (0.883, 0.957)	0.892 (0.855, 0.929)	5.711 (4.463, 6.959)
SU	47.5	0.5	0.940 (0.875, 1.005)	0.929 (0.839, 1.020)	6.196 (4.642, 7.750)	0.920 (0.883, 0.958)	0.892 (0.845, 0.939)	5.751 (4.517, 6.984)
OneR	51.4	0.5	0.943 (0.881, 1.005)	0.933 (0.849, 1.016)	6.286 (4.883, 7.688)	0.923 (0.889, 0.958)	0.895 (0.862, 0.928)	5.856 (4.777, 6.935)
SVM-RFE	87.9	2190.3	0.952 (0.884, 1.020)	**0.940** (0.860, 1.020)	**6.465** (4.965, 7.965)	0.932 (0.883, 0.980)	**0.900** (0.865, 0.935)	6.108 (4.692, 7.524)
LR-RFE	91.4	2190.4	0.951 (0.879, 1.023)	0.940 (0.858, 1.022)	6.437 (4.854, 8.019)	0.931 (0.880, 0.982)	0.899 (0.859, 0.939)	6.088 (4.614, 7.562)
RF-RFE	96.8	2190.2	**0.952** (0.881, 1.022)	0.940 (0.859, 1.020)	6.454 (4.916, 7.991)	**0.932** (0.881, 0.983)	0.900 (0.860, 0.939)	**6.114** (4.626, 7.603)
none	158.0	0.0	0.950 (0.882, 1.018)	0.940 (0.863, 1.016)	6.441 (5.030, 7.853)	0.931 (0.880, 0.982)	0.899 (0.861, 0.936)	6.075 (4.562, 7.588)

Source: author

There are two evident analytic perspectives arising from co-occurrence matrix, (1) feature importance across different selection procedures and (2) underlying similarity amongst results of feature selection schemes.

Considering the former perspective (1), three diverse groups of impact on the target variable are identified by the row-wise dendrogram. The bottom cluster consists of just one element – international_plan, which is recognized to be very important by all selection schemes; the middle cluster contains three elements – total_day_charge, number_customer_service_calls, total_day_minutes, that are also observed to be important indicators of customer's propensity to churn; the structure of the upper cluster is rather ambiguous, except for area_code element which is generally omitted.

From the latter perspective (2), three distinct groups of feature structures are identified by the column-wise dendrogram. The left cluster contains multivariate filter selection methods and Fischer's score; the middle cluster consists of EBM schemes and OneR; the right cluster is reserved for RFE procedures. The underlying similarity amongst selection schemes appears to be driven by both number and structure of included features; this is supported by the internal coherence of clusters considering the

performance of classification learner (see Tab. 3), albeit Consistency method does exhibit different behavior.

To outline a prediction performance of individual classification learners, mean point estimates across algorithms and selection/no selection schemes are displayed in Tab.4; the best indicators are depicted in bold; 95 % confidence intervals for underlying distributions are constructed.

RF algorithm presents superior performance with very low bias and acceptable variance across all performance measures; however, the drop in test performance might be a sign of overfitting. LOGIT method, on the other hand, displays higher bias and very low variance as a consequence of regularization. CIT and SVM learners exhibit akin performance with low bias and moderate variance.

To examine the behavior of classification learners further, another CIT model is built on top of the pipeline results; the response variable is top-decile lift measured on the test set, explanatory variables are feature selection scheme and classification method. The motivation for analyzing test TDL comes from its link to retention campaign profit dynamics (see Verbeke et al., 2012). The tree structure is charted in Appendix 1.; it becomes apparent that a feature selection procedure does not lead to significant improvement of the performance metric when combined with classification learners with embedded feature selection. This observation is supported by terminal nodes 12 (CIT), 18 (RF) and 23 (LOGIT) which blend learner's performance with and without feature selection. SVM learner, however, displays leap in performance when coupled with feature selection scheme. This conclusion is backed by comparison of boxplot charts in terminal node 10 (Consistency, EBMs, OneR) or 12 (RFE) with terminal node 15 (no feature selection scheme).

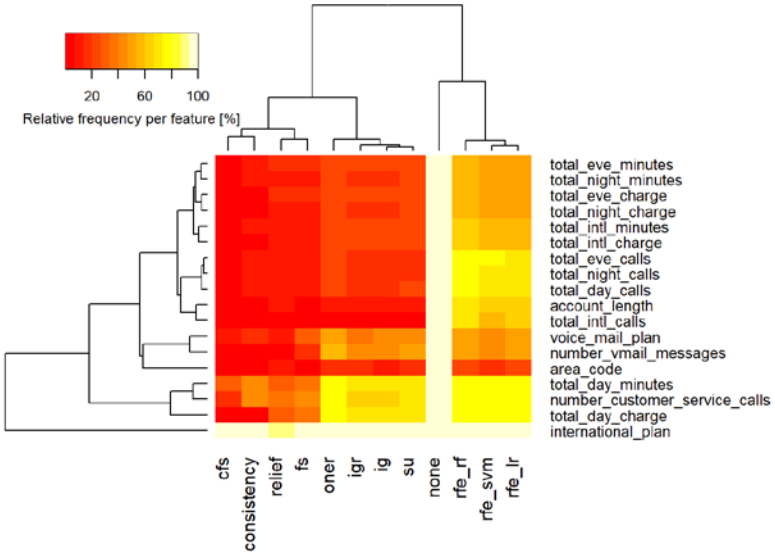


Figure 5. Scaled co-occurrence matrix for selection scheme-feature pairs, Source: author

The subsequent dimension of analysis comprises of time complexity of classification learners as a function of a number of explanatory variables (n). The empirical relationships are exposed by locally estimated scatterplot smoothing (LOESS) and depicted in Fig. 6.

Table 4. Classification performance indicators aggregated by the classification method

classification method	classification runtime [s]	train ACC (95 % CI)	Train AUC (95 % CI)	Train TDL (95 % CI)	test ACC (95 % CI)	test AUC (95 % CI)	test TDL (95 % CI)
LOGIT	21.8	0.893 (0.865, 0.920)	0.874 (0.814, 0.934)	4.891 (3.727, 6.054)	0.890 (0.864, 0.917)	0.868 (0.808, 0.928)	4.782 (3.647, 5.917)
CIT	52.2	0.941 (0.908, 0.974)	0.917 (0.865, 0.970)	6.335 (5.228, 7.442)	0.925 (0.891, 0.959)	0.879 (0.824, 0.933)	5.832 (4.596, 7.067)
SVM	244.3	0.939 (0.898, 0.981)	0.919 (0.865, 0.973)	6.315 (5.064, 7.565)	0.922 (0.895, 0.950)	0.897 (0.851, 0.943)	5.806 (4.855, 6.758)
RF	361.7	**0.980** (0.941, 1.020)	**0.996** (0.975, 1.017)	**6.965** (6.452, 7.478)	**0.940** (0.899, 0.981)	**0.906** (0.861, 0.951)	**6.318** (5.013, 7.623)

Source: author

From asymptotic perspective there appear to be three classes of behavior; (1) there is no clear relationship between number of features and classification runtime, suggesting complexity of $O(1)$, LOGIT flat line indicates such a nature; (2) there seems to be linear relationship between number of explanatory variables and classification runtime, indicating complexity of $O(n)$, this appears to be valid for SVM and RF models; (3) there is quadratic relationship between number of included variables and classification runtime, implying complexity of $O(n^2)$, this behavior fits the shallow convex curvature of CIT arc. RF LOESS, however, shows the systematic residual pattern in the middle and right sections of the figure; the observed phenomenon is induced by hyperparameter search step (sensitivity of a weak learner to a number of predictors and its depth).

6 Conclusions and future work

In an environment with steep data growth, it becomes inevitably hard to identify useful patterns and extract relevant knowledge. Thus, the goal of this paper is to examine the explanatory variable selection procedure in customer churn domain, specifically (1) its effect on prediction performance of a classification learner; (2) its behavior across explanatory variables; (3) a link between the number of included variables and classification runtime. The general topic is examined using an original experimental setup and utilizes publicly available dataset.

We witness slight improvement in learner's prediction performance when combined RFE selection, although the difference is not found statistically significant. From another viewpoint, RFE schemes allow us to reduce the number of features by ~ 40 % while retaining the same level of classification performance as with full-featured dataset. Consistency, EBM and OneR methods present notable behavior when heavily reducing the number of features while (1) being almost on par with RFE schemes across all performance measures and (2) being computationally less demanding.

When examining underlying feature importance across different selection schemes (see Fig. 5), international_plan, total_day_charge, number_customer_service_calls and total_day_minutes are recognized as important to the churn event; relevance of other features is inconclusive, except for area_code which is generally disregarded. From the perspective of business enterprise, the aforementioned findings may represent an invaluable insight into customer behavior. The latent similarity amongst results of feature selection procedures seem to be induced by number and structure of retained variables (see Fig. 5); the observation is supported by the internal coherence of clusters considering the performance of classification learner (see Tab. 3), albeit Consistency method does conduct adversely.

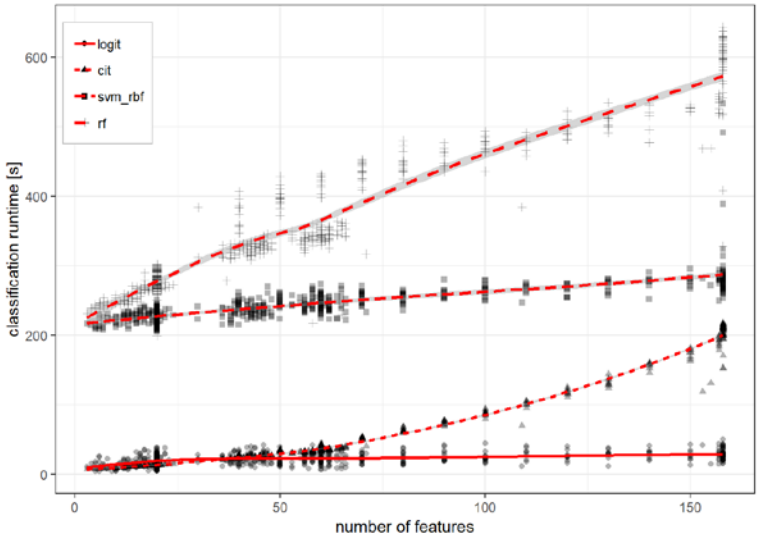


Figure 6. LOESS approximation of classification learner's runtime as a function of a number of included variables, Source: author

Considering the overall performance of classification learners, RFs exhibit superior behavior across all metrics. LOGIT learners distinct with higher bias and very low variance both of which are induced by regularization. CIT and SVM algorithms show comparable performance with low bias and moderate variance (see Tab. 4). We exploit a link between classifier's ability to generalize and feature selection procedure through the CIT model (see Appendix 1.). It becomes evident that incorporation of selection scheme does not improve performance metric when combined with classification learners with embedded feature selection. On the other hand, practitioners and researchers can tackle performance vs runtime trade-off with explicitly including selection scheme into machine learning pipeline; more specifically, by combining classifier with runtime sensitive to a number of features (CIT, SVM, RF) with efficient and computationally cheap univariate filter procedure (EBM, OneR). We can notice comparable benefits in EBM + SVM setup, which reduces computational runtime by ~ 15 % and improves test set TDL by ~ 5 % when compared to none + SVM setup (see Appendix 2.).

To illustrate the relevance of other parts of machine learning solution, we compare obtained results with selected research papers which utilize the same dataset, although their primary goals do not involve feature selection. We achieved performance comparable with Verbeke et al. (2012), the main discrepancy appears amongst LOGIT models where our incorporation of interaction features in data processing step leads to increase in test TDL by a factor of ~ 1.5. On the other hand, works of Vafeiadis et al. (2015) and Mehreen et al. (2017) exploit concepts of meta-learning which lead to increase in test ACC by ~ 5-10 % when compared to our endeavors.

As for future research of selection procedures in customer churn domain, we suggest considering more datasets and conceptually diverse classification learners. To explicitly address the trade-off between a number of features and information retained, multi-objective optimization might be leveraged in novel types of selection procedures. Another possible direction for research involves feature selection ensembles; meta-learning selection based on votes of multiple selection methods. From the perspective of the enterprise, adjusting feature selection procedures to business objectives in order to analyze retention drivers in profit perspective might be also a topic of interest.

Literature:

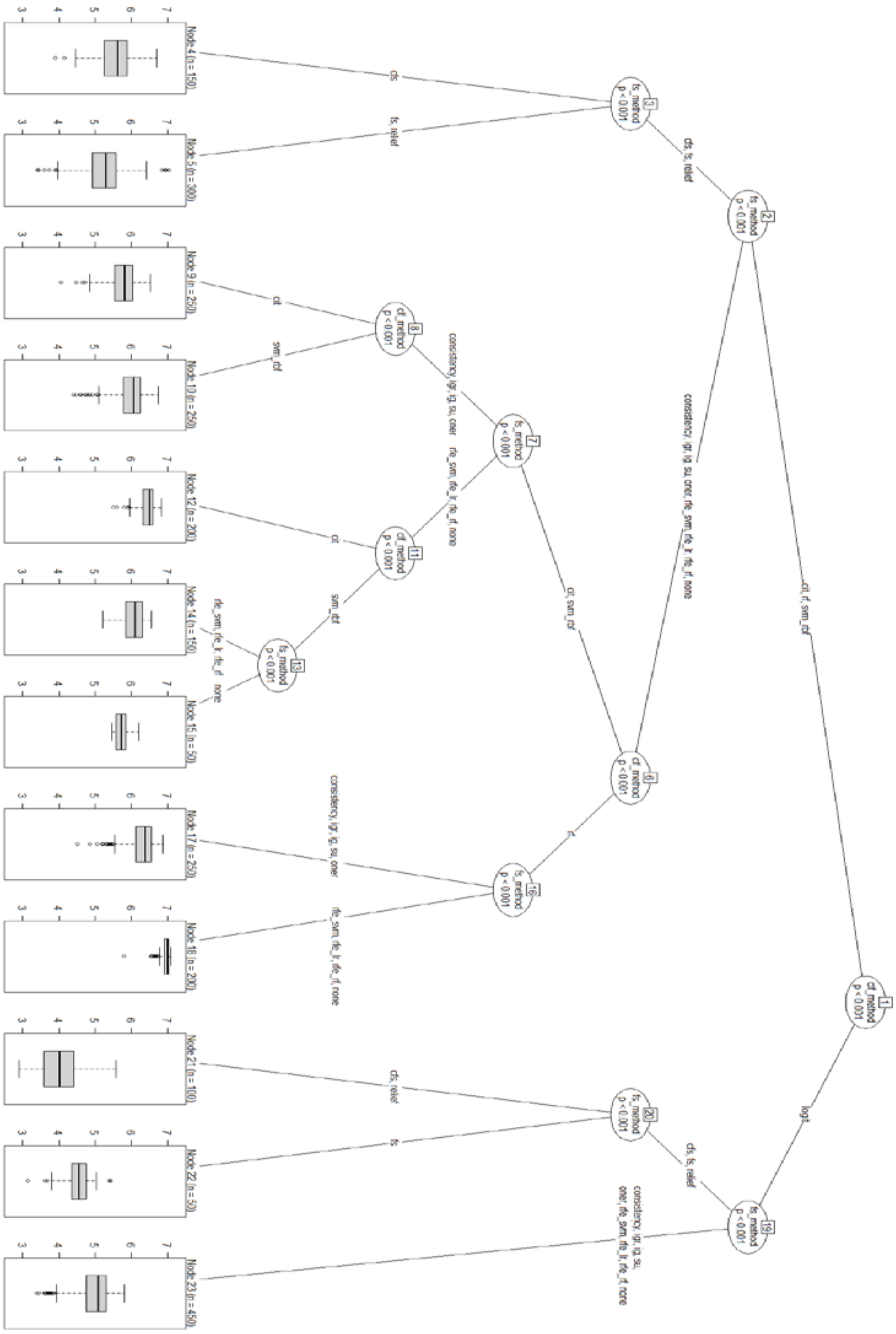
1. Aggarwal, C.C., 2014. Data classification: algorithms and applications, Boca Raton: Taylor & Francis.
2. Arauzo-Azofra, A., Benitez, J.M. & Castro, J.L., 2008. Consistency measures for feature selection. Journal of Intelligent Information Systems, 30(3), pp.273-292. Available at: http://link.springer.com/10.1007/s10844-007-0037-0.
3. Bagherzadeh-Khiabani, F. et al., 2016. A tutorial on variable selection for clinical prediction models: feature selection methods in data mining could improve the results. Journal of Clinical Epidemiology, 71, pp.76-85. Available at: https://linkinghub.elsevier.com/retrieve/pii/S0895435615004667
4. Bolón-Canedo, V., Sánchez-Maroño, N. & Alonso-Betanzos, A., 2013. A review of feature selection methods on synthetic

data. Knowledge and Information Systems, 34(3), pp.483-519. Available at: http://link.springer.com/10.1007/s10115-012-0487-8.
5. Bradley, A.P., 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognition, 30(7), pp.1145-1159. Available at: https://linkinghub.elsevier.com/retrieve/pii/S0031320396001422
6. Breiman, L., 2001. Random Forests. Machine Learning, 45(1), pp.5-32. Available at: http://link.springer.com/10.1023/A:1010933404324.
7. Dash, M. & Liu, H., 2003. Consistency-based search in feature selection. Artificial Intelligence, 151(1-2), pp.155-176. Available at: https://linkinghub.elsevier.com/retrieve/pii/S0004370203000791.
8. Duda, R.O., Hart, P.E. & Stork, D.G., 2012. Pattern Classification, Wiley. Available at: https://books.google.cz/books?id=Br33IRC3PkQC.
9. Fan, R.-E. et al., 2008. LIBLINEAR: A Library for Large Linear Classification. Journal of Machine Learning Research, 9, pp.1871-1874.
10. Gilhan, K. et al., 2010. An MLP-based feature subset selection for HIV-1 protease cleavage site analysis. Artificial Intelligence in Medicine, 48(2-3), pp.83-89. Available at: https://linkinghub.elsevier.com/retrieve/pii/S0933365709001031
11. Gronwald, K.D., 2017. Integrated Business Information Systems: A Holistic View of the Linked Business Process Chain ERP-SCM-CRM-BI-Big Data: A Holistic View of the Linked Business Process Chain ERP-SCM-CRM-BI-Big Data, Springer Berlin Heidelberg. Available at: https://books.google.cz/books?id=mSYmDwAAQBAJ.
12. Gupta, S., Lehmann, D.R. & Stuart, J.A., 2004. Valuing Customers. SSRN Electronic Journal. Available at: http://www.ssrn.com/abstract=459595.
13. Guyon, I. et al., 2002. Gene selection for cancer classification using support vector machines. Machine Learning, 46(1/3), pp.389-422. Available at: http://link.springer.com/10.1023/A:1012487302797.
14. Hall, M.A., 1999. Correlation-based feature selection for machine learning. Ph.D. thesis. Hamilton, New Zeland.
15. Hand, D.J., 2009. Measuring classifier performance: a coherent alternative to the area under the ROC curve. Machine Learning, 77(1), pp.103-123. Available at: http://link.springer.com/10.1007/s10994-009-5119-5.
16. Hothorn, T., Hornik, K. & Zeileis, A., 2006. Unbiased Recursive Partitioning: A Conditional Inference Framework. Journal of Computational and Graphical Statistics, 15(3), pp.651-674. Available at: http://www.tandfonline.com/doi/abs/10.1198/106186006X133933.
17. Hothorn, T. & Zeileis, A., 2015. Partykit: A Modular Toolkit for Recursive Partitioning in R. Journal of Machine Learning Research, 16, pp.3905–3909. Available at: http://jmlr.org/papers/volume16/hothorn15a/hothorn15a.pdf.
18. Chu, C. et al., 2012. Does feature selection improve classification accuracy? Impact of sample size and feature selection on classification using anatomical magnetic resonance images. NeuroImage, 60(1), pp.59-70. Available at: https://linkinghub.elsevier.com/retrieve/pii/S1053811911013486
19. Jin, C. & Wang, L., 2012. Dimensionality dependent PAC-Bayes margin bound. In Advances in Neural Information Processing Systems 25. Montreal, Canada: Curran Associates, pp. 1034-1042.
20. Karatzoglou, A. et al., 2004. Kernlab - An S4 Package for Kernel Methods in R. Journal of Statistical Software, 11(9). Available at: http://www.jstatsoft.org/v11/i09/.
21. Kononenko, I., 1994. Estimating attributes: Analysis and extensions of RELIEF. Machine Learning: ECML-94, pp.171-182. Available at: http://link.springer.com/10.1007/3-540-57868-4_57.
22. Kuhn, M., 2008. Building Predictive Models in R Using the caret Package. Journal of Statistical Software, 28(5). Available at: http://www.jstatsoft.org/v28/i05/.
23. Mehreen, A. et al., 2017. MCS: Multiple classifier system to predict the churners in the telecom industry. In 2017 Intelligent Systems Conference (IntelliSys). London, Great Britain: IEEE, pp. 678-683. Available at: http://ieeexplore.ieee.org/document/8324367/.
24. Powers, D.M.W., 2011. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. International Journal of Machine Learning Technology, 2(1), pp.37-63.
25. Shakil Pervez, M. & Md. Farid, D., 2015. Literature Review of Feature Selection for Mining Tasks. International Journal of Computer Applications, 116(21), pp.30-33. Available at: http://research.ijcaonline.org/volume116/number21/pxc3902829.pdf.
26. Spanoudes, P. & Nguyen, T., 2017. Deep Learning in Customer Churn Prediction: Unsupervised Feature Learning on Abstract Company Independent Feature Vectors. ArXiv:1703.03869 [cs, stat]. Available at: http://arxiv.org/abs/1703.03869.
27. Subramanya, K.B. & Somani, A., 2017. Enhanced feature mining and classifier models to predict customer churn for an E-retailer. In 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence. Noida, India: IEEE, pp. 531-536. Available at: http://ieeexplore.ieee.org/document/7943208/.
28. Torkzadeh, G., Chang, J.C.-J. & Hansen, G.W., 2006. Identifying issues in customer relationship management at Merck-Medco. Decision Support Systems, 42(2), pp.1116-1130. Available at: https://linkinghub.elsevier.com/retrieve/pii/S0167923605001491.
29. Vafeiadis, T. et al., 2015. A comparison of machine learning techniques for customer churn prediction. Simulation Modelling Practice and Theory, 55, pp.1-9. Available at: https://linkinghub.elsevier.com/retrieve/pii/S1569190X15000386.
30. Verbeke, W. et al., 2012. New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. European Journal of Operational Research, 218(1), pp.211-229. Available at: https://linkinghub.elsevier.com/retrieve/pii/S0377221711008599.
31. Vijaya, J. & Sivasankar, E., 2018. Computing efficient features using rough set theory combined with ensemble classification techniques to improve the customer churn prediction in telecommunication sector. Computing, 100(8), pp.839-860. Available at: http://link.springer.com/10.1007/s00607-018-0633-6.
32. Wright, M.N. & Ziegler, A., 2017. Ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. Journal of Statistical Software, 77(1). Available at: http://www.jstatsoft.org/v77/i01/.
33. Xiao, J. et al., 2015. Feature-selection-based dynamic transfer ensemble model for customer churn prediction. Knowledge and Information Systems, 43(1), pp.29-51. Available at: http://link.springer.com/10.1007/s10115-013-0722-y.
34. Zhu, Z., Ong, Y.-S. & Zurada, J.M., 2010. Identification of Full and Partial Class Relevant Genes. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 7(2), pp.263-277. Available at: http://ieeexplore.ieee.org/document/4653480/.

Primary Paper Section: A

Secondary Paper Section: AE, BB, IN

Appendix 1. Condition inference tree describing test TDL performance as a function of a feature selection procedure and classification method, Source: author



Appendix 2. Classification performance indicators aggregated per feature selection and classification method, Source: author

feature selection method	classification method	train ACC (95 % CI)	train AUC (95 % CI)	train TDL (95 % CI)	test ACC (95 % CI)	test AUC (95 % CI)	test TDL (95 % CI)
CFS	LOGIT	0.872 (0.856, 0.888)	0.842 (0.802, 0.882)	4.013 (2.963, 5.063)	0.872 (0.853, 0.890)	0.837 (0.790, 0.884)	3.954 (2.753, 5.155)
CFS	CIT	0.930 (0.904, 0.956)	0.909 (0.857, 0.960)	5.964 (5.022, 6.907)	0.910 (0.890, 0.931)	0.865 (0.821, 0.910)	5.333 (4.477, 6.189)
CFS	SVM	0.924 (0.893, 0.956)	0.904 (0.870, 0.937)	5.874 (4.823, 6.926)	0.917 (0.892, 0.941)	0.894 (0.871, 0.917)	5.609 (4.817, 6.401)
CFS	RF	0.965 (0.927, 1.003)	0.990 (0.966, 1.014)	6.858 (6.215, 7.501)	0.922 (0.893, 0.951)	0.900 (0.878, 0.922)	5.788 (4.833, 6.743)
Consistency	LOGIT	0.890 (0.872, 0.907)	0.871 (0.848, 0.895)	4.827 (3.996, 5.658)	0.888 (0.869, 0.906)	0.868 (0.838, 0.898)	4.760 (3.908, 5.612)
Consistency	CIT	0.939 (0.924, 0.955)	0.925 (0.893, 0.956)	6.318 (5.800, 6.836)	0.922 (0.904, 0.939)	0.880 (0.847, 0.914)	5.759 (5.161, 6.358)
Consistency	SVM	0.943 (0.915, 0.971)	0.920 (0.894, 0.947)	6.457 (5.636, 7.277)	0.928 (0.906, 0.950)	0.900 (0.879, 0.922)	5.993 (5.282, 6.704)
Consistency	RF	0.984 (0.967, 1.001)	0.999 (0.994, 1.003)	7.057 (7.004, 7.109)	0.937 (0.919, 0.954)	0.910 (0.892, 0.928)	6.271 (5.689, 6.852)
FS	LOGIT	0.885 (0.863, 0.907)	0.856 (0.790, 0.923)	4.553 (3.682, 5.424)	0.884 (0.863, 0.904)	0.851 (0.765, 0.937)	4.507 (3.642, 5.372)
FS	CIT	0.922 (0.893, 0.951)	0.898 (0.828, 0.968)	5.673 (4.567, 6.780)	0.907 (0.878, 0.936)	0.862 (0.778, 0.946)	5.158 (4.041, 6.276)
FS	SVM	0.918 (0.881, 0.955)	0.891 (0.832, 0.950)	5.635 (4.364, 6.906)	0.910 (0.881, 0.938)	0.875 (0.802, 0.948)	5.331 (4.213, 6.449)
FS	RF	0.955 (0.908, 1.001)	0.982 (0.934, 1.031)	6.668 (5.707, 7.629)	0.916 (0.881, 0.950)	0.885 (0.814, 0.956)	5.496 (4.246, 6.746)
Relief	LOGIT	0.875 (0.856, 0.895)	0.825 (0.728, 0.923)	4.052 (3.072, 5.032)	0.875 (0.856, 0.893)	0.824 (0.737, 0.910)	4.016 (3.072, 4.961)
Relief	CIT	0.919 (0.887, 0.951)	0.889 (0.799, 0.978)	5.564 (4.343, 6.784)	0.902 (0.874, 0.930)	0.848 (0.763, 0.933)	4.955 (3.788, 6.122)
Relief	SVM	0.911 (0.876, 0.945)	0.881 (0.810, 0.951)	5.435 (4.245, 6.624)	0.903 (0.879, 0.927)	0.863 (0.789, 0.936)	5.143 (4.290, 5.996)
Relief	RF	0.955 (0.914, 0.997)	0.987 (0.967, 1.007)	6.726 (5.983, 7.468)	0.911 (0.879, 0.943)	0.875 (0.801, 0.949)	5.358 (4.215, 6.500)
IGR	LOGIT	0.896 (0.869, 0.923)	0.878 (0.824, 0.931)	5.000 (3.894, 6.106)	0.894 (0.868, 0.921)	0.875 (0.818, 0.932)	4.890 (3.713, 6.066)
IGR	CIT	0.937 (0.914, 0.960)	0.917 (0.875, 0.959)	6.225 (5.427, 7.024)	0.921 (0.898, 0.944)	0.880 (0.834, 0.926)	5.696 (4.910, 6.481)
IGR	SVM	0.938 (0.897, 0.979)	0.916 (0.861, 0.970)	6.305 (5.025, 7.585)	0.923 (0.893, 0.954)	0.896 (0.854, 0.939)	5.840 (4.816, 6.864)
IGR	RF	0.978 (0.943, 1.012)	0.996 (0.980, 1.012)	6.967 (6.460, 7.474)	0.934 (0.904, 0.965)	0.904 (0.862, 0.946)	6.183 (5.154, 7.213)
IG	LOGIT	0.898 (0.875, 0.920)	0.883 (0.852, 0.914)	5.064 (4.163, 5.965)	0.896 (0.874, 0.918)	0.879 (0.845, 0.913)	4.973 (4.051, 5.896)
IG	CIT	0.939 (0.920, 0.958)	0.921 (0.894, 0.948)	6.303 (5.692, 6.915)	0.921 (0.901, 0.941)	0.882 (0.846, 0.917)	5.717 (4.957, 6.477)
IG	SVM	0.941 (0.905, 0.978)	0.921 (0.885, 0.957)	6.411 (5.302, 7.519)	0.925 (0.900, 0.951)	0.900 (0.878, 0.922)	5.905 (5.026, 6.783)
IG	RF	0.980 (0.950,	0.997 (0.986,	7.005 (6.727,	0.936 (0.911,	0.908 (0.888,	6.249 (5.408,

		1.009)	1.008)	7.283)	0.961)	0.927)	7.090)
SU	LOGIT	0.899 (0.879, 0.918)	0.883 (0.845, 0.922)	5.122 (4.378, 5.867)	0.897 (0.876, 0.917)	0.880 (0.832, 0.927)	5.036 (4.218, 5.854)
SU	CIT	0.938 (0.917, 0.959)	0.915 (0.868, 0.962)	6.261 (5.511, 7.012)	0.922 (0.900, 0.944)	0.883 (0.837, 0.929)	5.737 (4.915, 6.558)
SU	SVM	0.940 (0.905, 0.976)	0.920 (0.879, 0.961)	6.392 (5.258, 7.526)	0.926 (0.898, 0.953)	0.899 (0.864, 0.935)	5.937 (5.041, 6.834)
SU	RF	0.983 (0.952, 1.014)	0.998 (0.987, 1.009)	7.008 (6.647, 7.370)	0.937 (0.910, 0.964)	0.907 (0.872, 0.941)	6.293 (5.435, 7.152)
OneR	LOGIT	0.902 (0.884, 0.920)	0.889 (0.868, 0.909)	5.259 (4.665, 5.853)	0.900 (0.882, 0.917)	0.885 (0.861, 0.908)	5.143 (4.531, 5.756)
OneR	CIT	0.940 (0.924, 0.957)	0.920 (0.892, 0.947)	6.354 (5.802, 6.906)	0.924 (0.907, 0.941)	0.884 (0.850, 0.919)	5.837 (5.238, 6.437)
OneR	SVM	0.944 (0.916, 0.972)	0.923 (0.893, 0.953)	6.483 (5.630, 7.336)	0.928 (0.910, 0.946)	0.902 (0.883, 0.921)	6.023 (5.446, 6.599)
OneR	RF	0.986 (0.964, 1.008)	0.999 (0.993, 1.005)	7.046 (6.862, 7.230)	0.941 (0.924, 0.959)	0.911 (0.892, 0.929)	6.422 (5.869, 6.975)
SVM-RFE	LOGIT	0.900 (0.879, 0.921)	0.891 (0.870, 0.911)	5.211 (4.614, 5.807)	0.896 (0.877, 0.914)	0.882 (0.863, 0.901)	5.061 (4.462, 5.659)
SVM-RFE	CIT	0.956 (0.942, 0.970)	0.929 (0.898, 0.959)	6.806 (6.438, 7.174)	0.942 (0.927, 0.956)	0.890 (0.864, 0.915)	6.405 (5.966, 6.845)
SVM-RFE	SVM	0.958 (0.936, 0.980)	0.942 (0.918, 0.966)	6.783 (6.295, 7.271)	0.931 (0.913, 0.949)	0.911 (0.895, 0.928)	6.073 (5.482, 6.664)
SVM-RFE	RF	0.993 (0.977, 1.008)	1.000 (0.999, 1.001)	7.061 (7.061, 7.061)	0.960 (0.946, 0.973)	0.918 (0.899, 0.937)	6.894 (6.486, 7.303)
LR-RFE	LOGIT	0.896 (0.876, 0.916)	0.888 (0.870, 0.905)	5.112 (4.531, 5.693)	0.893 (0.873, 0.913)	0.879 (0.858, 0.901)	4.983 (4.389, 5.577)
LR-RFE	CIT	0.957 (0.943, 0.971)	0.931 (0.896, 0.965)	6.826 (6.425, 7.228)	0.941 (0.925, 0.958)	0.888 (0.849, 0.927)	6.402 (5.896, 6.909)
LR-RFE	SVM	0.956 (0.931, 0.981)	0.940 (0.916, 0.965)	6.747 (6.201, 7.293)	0.930 (0.914, 0.946)	0.910 (0.892, 0.928)	6.056 (5.527, 6.584)
LR-RFE	RF	0.995 (0.981, 1.009)	1.000 (0.999, 1.000)	7.061 (7.061, 7.061)	0.960 (0.948, 0.972)	0.918 (0.899, 0.936)	6.911 (6.609, 7.213)
RF-RFE	LOGIT	0.898 (0.877, 0.919)	0.890 (0.869, 0.910)	5.170 (4.562, 5.778)	0.893 (0.873, 0.914)	0.880 (0.860, 0.899)	4.992 (4.408, 5.575)
RF-RFE	CIT	0.958 (0.948, 0.969)	0.929 (0.900, 0.958)	6.854 (6.556, 7.152)	0.944 (0.931, 0.956)	0.890 (0.853, 0.926)	6.463 (6.089, 6.837)
RF-RFE	SVM	0.956 (0.931, 0.981)	0.941 (0.918, 0.964)	6.730 (6.177, 7.283)	0.930 (0.913, 0.946)	0.911 (0.893, 0.929)	6.049 (5.514, 6.584)
RF-RFE	RF	0.995 (0.982, 1.007)	1.000 (1.000, 1.000)	7.061 (7.061, 7.061)	0.961 (0.953, 0.970)	0.918 (0.901, 0.936)	6.954 (6.765, 7.143)
none	LOGIT	0.902 (0.883, 0.921)	0.896 (0.880, 0.913)	5.306 (4.743, 5.868)	0.897 (0.877, 0.916)	0.880 (0.860, 0.900)	5.071 (4.431, 5.711)
none	CIT	0.959 (0.944, 0.974)	0.929 (0.897, 0.960)	6.873 (6.459, 7.287)	0.945 (0.931, 0.960)	0.892 (0.854, 0.930)	6.515 (6.038, 6.992)
none	SVM	0.944 (0.923, 0.966)	0.933 (0.912, 0.955)	6.526 (5.998, 7.055)	0.919 (0.909, 0.930)	0.903 (0.886, 0.920)	5.717 (5.383, 6.051)
none	RF	0.995 (0.981, 1.009)	1.000 (1.000, 1.000)	7.061 (7.061, 7.061)	0.963 (0.956, 0.970)	0.919 (0.901, 0.937)	6.996 (6.873, 7.119)