FRI - Feature Relevance Intervals for Interpretable and Interactive Data Exploration

1st Lukas Pfannschmidt
DiDy & Machine learning group
Bielefeld University
Bielefeld, Germany
lpfannschmidt@techfak.uni-bielefeld.de

2nd Christina Gpfert
Machine learning group
Bielefeld University
Bielefeld, Germany
cgoepfert@techfak.uni-bielefeld.de

3rd Ursula Neumann
Dep. of Mathematics and Computer Science
University of Marburg
Marburg, Germany
ursula.neumann@staff.uni-marburg.de

4th Dominik Heider
Dep. of Mathematics and Computer Science
University of Marburg
Marburg, Germany
dominik.heider@uni-marburg.de

5th Barbara Hammer
Machine learning group
Bielefeld University
Bielefeld, Germany
bhammer@techfak.uni-bielefeld.de

***Abstract*—Most existing feature selection methods are insufficient for analytic purposes as soon as high dimensional data or redundant sensor signals are dealt with since features can be selected due to spurious effects or correlations rather than causal effects. To support the finding of causal features in biomedical experiments, we hereby present FRI, an open source *Python* library that can be used to identify all-relevant variables in linear classification and (ordinal) regression problems. Using the recently proposed feature relevance method, FRI is able to provide the base for further general experimentation or in specific can facilitate the search for alternative biomarkers. It can be used in an interactive context, by providing model manipulation and visualization methods, or in a batch process as a filter method.**

***Index Terms*—global feature relevance, feature selection, interpretability, interactive biomarker discovery**

I. Background

In recent years, due to an increasing availability of highly sensitive biotechnologies as well as an increasing digitalization of biomedical diagnostics, one could observe a trend towards bigger and more complex machine learning models. These models are used successfully in medical diagnoses [1] [2] such as cancer prediction [3] using known biomarkers as input. When specific biomarkers are unknown, many learning models can also perform on nearly raw data without a preselection of variables. While such data representation often enables a high prediction accuracy, it is less suited if the purpose is data exploration and understanding of the underlying causal relationships. For the latter, insight into the model behavior and its relevant driving factors is necessary [4] and feature selection constitutes a first step to unravel the underlying relationships. But even for predictive models, the inference of sparse models can have a significant impact on the model performance, since it helps to mediate the curse of dimensionality thus leading to a better generalization ability and improved computational complexity. Because of this, there is a big need for methods to reveal relevant structure and function in the data itself.

Feature Selection (FS) constitutes one particularly prominent paradigm which enables the inference of sparse and interpretable prediction models [5]. The task is mostly undertaken in conjunction with model selection to improve predictive performance. Then the problem is often defined as finding the minimal subset of all features to achieve the best performance given some objective score. Commonly, one distinguishes filter, wrapper, and embedded methods. Filter methods are based on the information content of features as regards the output class label, disregarding the specific classification method. Hence they are particularly suited as a first screening technology for high dimensional data [6]. Wrapper approaches add (remove) features to a growing (shrinking) candidate set [7] while evaluating an inner model on a predefined metric. Embedded approaches use the internal model weights to find relevant features such as the *Lasso* [8] which enforces sparsity through its choice of regularization. Since they do not rely on iterative feature selection or weighting, embedded approaches have the benefit that they can effectively take into account interdependencies of groups of features. Further, they are specific to the used model.

In this contribution, we will focus on embedded methods for one particularly prominent group of models, namely linear classifiers or regression models, such as also addressed by the famous *Lasso*. While linear models can not applied to every problem, they are widely used in medicine [9] and the growing size of data makes their efficiency even more relevant. Furthermore, methods such as *Lasso* can be accompanied by formal guarantees on their ability to identify the true underlying features in the limit [10]. Yet, these guarantees do not hold if conditions as regards unique representability are violated. The latter is the case if features are high dimensional, highly correlated, or if there do exist different feature sets which yield the same prediction accuracy. In such cases, prediction

This contribution has been made possible through the German DFG research training group Computational Methods for the Analysis of the Diversity and Dynamics of Genomes (DiDy) GRK 1906/1.

accuracy in the limit still holds [11] yet the fact that *Lasso* can identify the true underlying features is violated. In particular, greedy and sparse approaches lead to instabilities of the selected feature set when correlated features are present. They remove similar features which could otherwise be grouped into potential functional units.

Finding a minimal subset (the usual objective of *Lasso* or related methods) is therefore not suited for giving a complete picture of all relevant features but only those which are sufficient to fulfill the prediction. To gain information about the importance of *all* features one needs to consider a more general measure: *feature relevance* [7] [12]. The membership of the minimal feature set as a binary measure of relevance is too narrow in most cases. *Kohavi et al.* [7] coined the term *all relevant feature selection* (ARFS) in the 90s and added further distinction of relevance. Additional to strongly relevant features, which need to be part of the candidate feature set to achieve the best performance, and irrelevant features which are not beneficial for the considered relationship at all, they also introduced the concept of weakly relevant features i.e. features which can contribute information to a model, but are not necessarily included in every good model.

Weakly relevant features are often highly correlated with each other. They are not limited to pairs but also bigger groups of associated features. Out of these, at least one has to be part of the candidate set. Having knowledge about a group of features which could fulfill the same role in a model can be very important in the design of diagnostic tests where the source of data can differ by the cost or invasiveness of acquisition. Explicit redundancies of feature relevance then enable a practitioner to avoid, e.g., expensive features if they can be substituted by others. Additionally, feature groups could induce novel biological relationships. This is especially useful in gene co-expression or metabolomics experiments where groups of functional units are common [13]. Interestingly, it has been shown that extensions of *Lasso* which take into account feature correlations such as group *Lasso* can lead to wrong results due to a selection bias caused by feature correlations [14].

Finding an optimal solution to the ARFS problem is computationally intractable [15] but approximations exist. In 2005 the *ElasticNet* [16] overcame some of the instability of sparse models [17] by using a combination of multiple different regularization terms which lead to better conservation of weakly relevant features. The *statistically equivalent signature (SES)* [18] approach proposes a technology which groups mutually equivalent features into groups out of which minimal feature subsets can be constructed. Another proposal called *stability selection* uses resampling for more robust selection especially in high dimensional problems [19] [20]. In 2010 *Boruta* [21] specifically focused on finding all relevant features using statistical testing of contrast features. Alternatives to the Boruta method are discussed and evaluated in [22], whereby Boruta was identified as best performing technology among the tested ones if used for different dimensionalities of the data. In 2017 *Neumann et al.* presented the *Ensemble Feature Selection* (EFS) method which combines multiple FS methods to remove individual biases and give aggregated feature relevance ranges for all of them [23], [24].

In this paper, we focus on the question of how to efficiently uncover a detailed view on the relevance of features in the case of possible feature redundancies. Thereby, we investigate linear models as a particularly relevant setting. The main contribution of this article is an accessible implementation of the *feature relevance interval* method (*FRI*) [25][1]. The *FRI* offers an efficient framework which assigns relevance intervals to features, rather than simple coefficient values. These intervals mirror the coefficient range of a feature when considering *all* possible models, hence offering detailed and complete information also in the case of feature redundancies. In this contribution, we offer an interactive software tool based on the mathematical framework as derived in [25] and we demonstrate its applicability in the context of biomedical data analysis. More precisely, we show that using these bounds, we can classify each feature into one of the three relevance classes. Furthermore, we can use these relevance bounds in visualizing the model which allows interpretability. Especially the information provided by the discrimination between strongly and weakly relevant features can help in biomedical applications. As an example we look at model design and biomarker discovery, where it could highlight elements which are crucial for the problem at hand while also providing alternative markers which have the same information.

The article is structured as follows: In chapter II we shortly recapitulate the theoretical background and give details of the implementation of the method using linear programming. Specifically in II-A1 we show how to obtain a baseline solution and give our definition of relevance bounds in II-A2. We then describe how we can classify each feature into three relevance groups and how to reduce false positives using a probe based threshold estimation in II-A3. Then we show how we can constrain the use of features to certain relevance values (II-B) to facilitate interactive data exploration and model design. In III-A we evaluate the method quantitatively using simulated (III-A1) and biomedical data (III-A2). We also provide a qualitative evaluation and example in III-B.

II. Implementation

A. Feature Relevance Intervals

For brevity, in the following definitions and experiments we only consider the task of binary classification, but the method and our implementation can be applied to linear regression and ordinal regression problems as well [26].

For a classification problem we observe data as $(x^1, y^1), \ldots, (x^n, y^n) \quad \in \mathbf{R}^d \times \{-1, 1\}$ with n different samples and d real-valued features which have been tied to a target or response y. We assume that all d features have been standardized at mean zero and standard deviation 1.

We propose an interactive pipeline, which enables the practitioner a detailed view of the relevance of features on

[1]Source code and package availabe at github.com/lpfann/fri.

the classification also in the case of multiple solutions or feature redundancies. More precisely, we propose an efficient framework which determines the relevance of features as regards all possible solutions of the classification problem, we demonstrate how this information can be used to identify all strongly as well as weakly relevant features, and we present a procedure to iteratively investigate feature combinations. Fig. 1 displays the processing pipeline of the proposed method.

1) Baseline Solution: It is common practice to evaluate the relevance of a feature for a given classification by means of the weights assigned to the feature by a linear classifier such as a support vector machine [27]. Thereby, sparsity can be emphasized by resorting to, e.g., *Lasso* or sparse SVM models [8] [28]. Yet, provided the solution is not unique, the resulting feature relevance is to some extent arbitrary, i.e. possibly rendering the model interpretation invalid. Here we propose an alternative: Instead of taking the weights of a single linear model as an approximation of feature relevances we look into using the complete model class of well-performing sparse linear classifiers. A model class is characterized by its similarity in the quality of the solution i.e. similar generalization ability as characterized by the size of the weight vector of the SVM [29] and similar training loss, which measures wrongly classified samples. Through the use of a class of models, we can approximate the global solution of the *ARFS* problem as shown in [25].

Assume we are interested in linear classifiers of the form $y \mapsto sgn(\omega^T x - b)$ where ω is the normal vector of the separating hyperplane, b denotes the bias and *sgn* refers to the sign function. First, we acquire a baseline solution to the problem using an l_1-regularized soft-margin SVM:

$$\begin{aligned}\left(\tilde{\omega}, \tilde{b}, \tilde{\xi}\right) \in \arg\min_{\omega,b,\xi} \|\omega\|_1 + C \cdot \sum_{i=1}^{n} \xi_i \\ \text{s.t. } y_i(\omega^\top x_i - b) \geq 1 - \xi_i, \forall i \\ \xi_i \geq 0, \forall i\end{aligned}$$

Through optimization, we acquire a model fully defined by the normal vector of a hyperplane ω and its offset from the origin b. Prediction of samples is based on the signed distance from the plane. As usual, ξ_i are slack variables to guarantee the feasibility of the constraint optimization problem in case of unavoidable classification errors. C is a regularization parameter which depends on the datasets distribution. We choose the parameter guided by 3-fold stratified cross-validation and the F_1 score weighted by each class support to account for class imbalances.

From the model with the best C, we obtain constraints for controlling the generalization error of equivalent models: the upper limit on the weight vector $\mu := \|\tilde{\omega}\|_1$ and error term $\rho := \sum_{i=1}^{n} \tilde{\xi}_i$. These values determine the class of equivalent classifiers, which consist of all SVM solutions (ω', b', ξ') such that $\|w'\|_1 \leq \mu$ and $\sum \xi'_i \leq \rho$. All these alternatives are considered equivalent since they show the same performance for the given classification task as the found solution. Hence all weight vectors associated with an equivalent solution are relevant to determine the relevance of a feature to the given classification problem.

2) Minimum and Maximum Bounds: Using these constraints we now define feature relevance bounds for every feature j independently i.e. we determine the interval of weight vectors which results if we take into account the weights of all possible equivalent classifiers. Mathematically speaking, we want to find extreme weight values for each feature given a similar error to our baseline. This can be done using linear programming.

For the lower bound, the lowest possible value of feature j, we define the problem

$$\begin{aligned}\text{minRel}((x_i, y_i)_{i=1}^n, j) : \min_{\omega,b,\xi} |\omega_j| \\ \text{s.t. } y_i(\omega^\top x_i - b) \geq 1 - \xi_i, \xi_i \geq 0 \forall i \\ \sum_{i=1}^{n} \xi_i \leq \rho \\ \|\omega\|_1 \leq (1+\delta) \cdot \mu.\end{aligned} \tag{1}$$

And the upper bound for j is defined as

$$\begin{aligned}\text{maxRel}((x_i, y_i)_{i=1}^n, j) : \max_{\omega,b,\xi} |\omega_j| \\ \text{s.t. constraints } (1) \text{ hold.}\end{aligned}$$

The optimization problems can be rewritten as linear optimization problems and solved in polynomial time [25] using appropriate solvers. To account for their numerical inaccuracies, which we encountered in our experiments, we also propose a relaxation factor $\delta = 0.001$ in (1) to allow minor deviations from μ. Given that all problems can be solved independently, our implementation makes use of parallel computation.

We end up with a real-valued matrix $\mathbb{RI} \in \mathbb{R}^{2\times d}$ which contains all pairs of relevance bounds. These intervals give an indication about the degree up to which a feature can or must be used in the classification and they can be visualized for interpretative purposes as seen in Fig. 2.

3) Feature Classification: By intuition one could use the following three rules to map feature relevances to their relevance class:

Strongly relevant A feature is strongly relevant when its lower relevance bound is bigger than zero. The model class defined by its prediction accuracy, is dependent on information from it.

Weakly relevant When two or more features are correlated, they can replace each other functionally in the model. These features are characterized by a lower bound equal to zero and an upper bound bigger than zero.

Irrelevant By definition, irrelevant features should have no measured relevance at all. Their upper (and lower) bound should therefore be zero.

In practice, the discrimination between relevant and irrelevant features is challenging. The use of slack variables in the overall model and thus our relevance bounds allow variation in the contribution of features. For relevance bounds specifically, even if feature j is independent we observe $\text{maxRel}_j > 0$. One

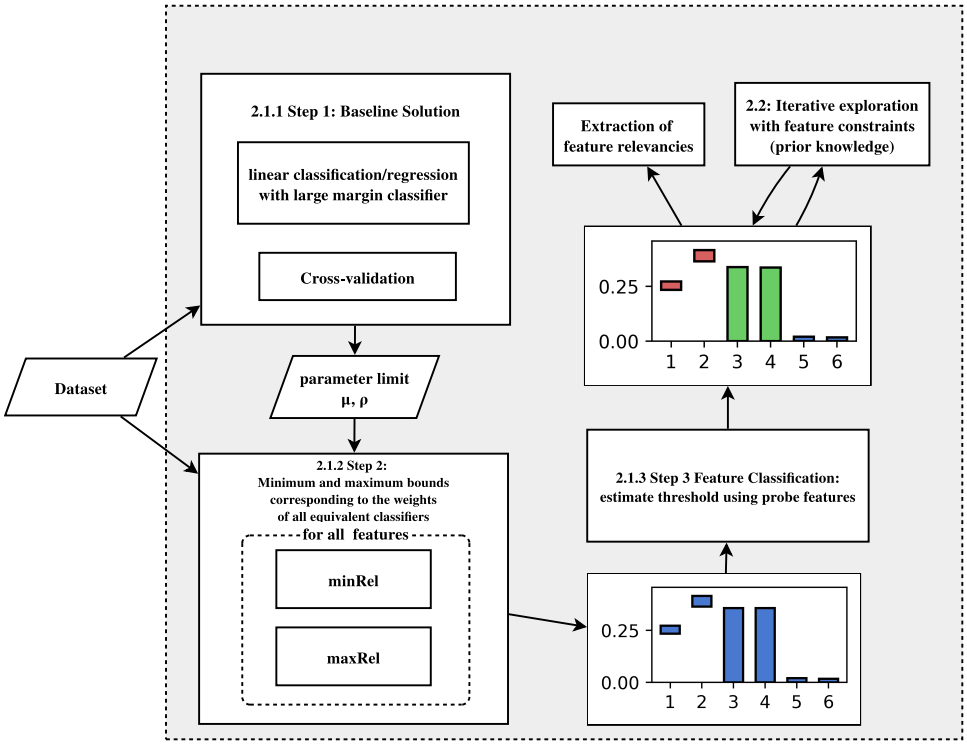

Fig. 1. Overview of *FRI*.

could introduce a naive data independent threshold to discriminate between noise and relevant features, but this would lead to bad precision or recall of features in most cases. Instead we try to estimate the distribution of relevances of noise features given the model constraints. We expect for a given model class the same amount of allowed variation in the relevances and therefore a normal distribution with an unknown mean and variance. We propose to estimate the distributions parameters and the corresponding prediction interval (PI) to obtain a data dependent threshold [30]. To estimate this noise distribution we use n permutated (p) input features from X i.e. for each j^p we compute maxRel$((\hat{x}_i, y_i)_{i=1}^n, j^p)$ where $j \notin \hat{x}$.

The prediction interval is then defined as

$$PI := \overline{P}_n \pm T_{n-1}(p) s_n \sqrt{1 + (1/n)}.$$

Here $\overline{P}_n$ denotes the sample mean and s_n the standard deviation and T represents the Student's t-distribution with $n-1$ degrees of freedom. The size of PI depends on parameter p, the expected probability that a new value is included in the interval. We propose default values of $p = 0.999$ for a low false positive rate and $n >= 50$ to achieve a stable distribution.

To classify a feature j as irrelevant we check if maxRel$_j \in PI$. To discriminate between weak and strong relevance we then additionally check the lower bound as described at the beginning of II-A3. An example of the feature classification can be seen in Fig. 2 where the relevance bars are colored according to their relevance class.

Our method provides the vector $\mathbb{ARR} \in \{0, 1, 2\}^d$, which encodes strongly (2), weakly (1) and irrelevant (0) features.

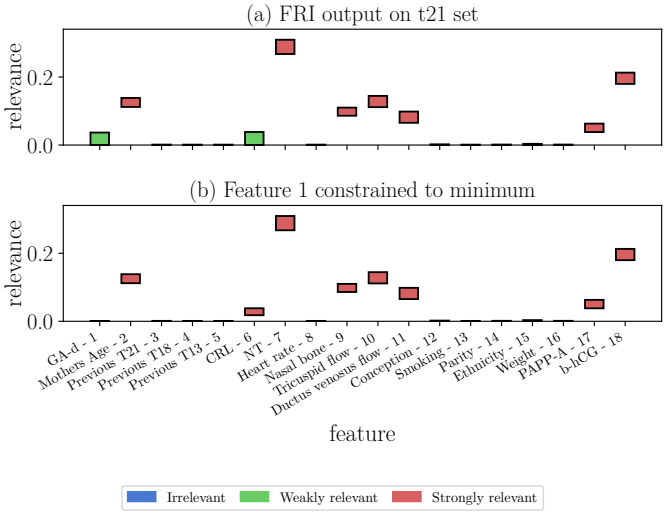

Fig. 2. Program output using the *t21* dataset visualizing relevance bounds for all features as colored boxes. Colors correspond to relevance classes assigned by *FRI*. (a) Shows program output without any constraints introduced by the user. (b) Shows output with feature 1 GA-d ("Gestation age in days") set to its minimum value.

B. Feature Constraints

The mathematical formalization of relevance intervals as introduced above also opens up the opportunity to integrate prior knowledge about feature relevances and to interactively explore good solutions by means of integrating additional bounds for specific features. More precisely by solving the

problem using linear programs, the addition of constraints is very easy. One way we leverage this is the possibility of adding relevance constraints to the optimization.

Given data $\mathbb{D} := (X^{n \times d}, y^n)$, feature set $\{\mathcal{S}\}, |\mathcal{S}| = d$ and feature $i \in \mathcal{S}$. We define an allowed relevance range for feature i: $\left[fc^i_{min}, fc^i_{max}\right]$ where $fc^i_\bullet \in \mathbb{R}_{\geq 0}$.

To compute relevance bounds, including individual feature constraints, we have to extend the set of existing constraints in the optimization from Step 2 in Section II-A. The minimum relevance bound with *constraints* is defined as

$$\text{minRel}\mathbf{C}(\mathbb{D}, j, fc) : \min_{\omega, b, \xi} |\omega_j|$$
$$\text{s.t. constraints } (1) \text{ hold and}$$
$$fc^k_{min} \geq |\omega_k| \geq fc^k_{max} \forall k \neq j.$$

The maximum bound is defined analogously to this. In the case of one value $fc^i_{min} = fc^i_{max}$ we consider feature i as fixed. To rewrite the new absolute term $|\omega_k|$ as a convex problem, we utilize the baseline solution $\tilde{\omega}$, which allows us to use the sign of the coefficient $\tilde{\omega}_k$ turning the non-convex absolute term into a simple convex one.

By changing the amount of contribution we allow for one feature, we can observe varying intervals for others and infer potential dependencies between them as in Fig. 2 (b). This is especially useful when designing a set of biomarkers. In our tool we provide the means to easily define ranges or values for all features. Presetting values can be based on the relevance intervals which give a first indication of importance. We also plan to integrate a function to automatically group features based on these constrained relevance bounds in the future.

III. Results

A. Feature Selection Evaluation

To evaluate the method in context, we run benchmarks against other established methods: *Boruta* as a representative of a method with statistical testing [31], *Ensemble Feature Selection* (EFS) [23], *ElasticNet* using an equal contribution of L_1 and L_2 regularization [16] and *stability selection* (SS) [19] [20]. We removed the Lasso from the comparison because of very similar performance to the EN. For all methods, the proposed default parameters are used. Hyperparameters are selected according to a cross validation scheme. For the ElasticNet, we choose the feature set depending on the coefficients c_i of the model where $c_i > 10^{-5}$ counts as selected.

We used two types of sets: generated sets where we had knowledge of the underlying ground truth and real world data stemming from medical studies.

1) Simulation data: All our simulation sets are sampled from a binary classification problem. To generate a multidimensional classification problem, we use a randomly generated prototype vector which defines a hyperplane. The defining features of this plane are strongly relevant. Now points are sampled in this feature space and the class is determined by the side of the hyperplane the points lie on. Weakly relevant features are constructed by replacing a feature of the original feature space with its linear combination. The elements of this combination are highly correlated and produce a set of redundant features. By removing the original feature and replacing it with those elements we achieve weak relevance by definition. Irrelevant features are sampled from a standard normal distribution. All simulation sets consist of 30 features and 500 samples. They differ in the density of the relevant feature space which is defined by the amount of strongly, weakly and irrelevant variables which are listed in Table I. According to these parameters 50 sets were generated per configuration and the following evaluation refers to the aggregated scores. *Sim1* and *Sim3* have a sparse relevant feature space while *Sim2* and *Sim4* are dense. Additionally, in *Sim1* and *Sim2* weakly relevant features are present, while they are missing completely in *Sim3* and *Sim4*. *Sim5* had all strongly relevant features removed.

TABLE I
Characteristics of simulated datasets. Each set consists of 30 features with 500 samples.

	Strongly relevant	Weakly relevant	Irrelevant
Sim1	4	4	22
Sim2	12	8	10
Sim3	4	0	26
Sim4	18	0	12
Sim5	0	20	10

Performance on these sets can give clues about the effectiveness of the considered feature selection strategy. Due to a known ground truth, we can explicitly evaluate the validity of the selected features. We focus on the all-relevant feature selection problem and we use the following measures to evaluate the match of the detected feature set and the known ground truth of all relevant features: precision and recall. Recall is defined by TP / (TP+FN) with TP = true positives and FN = false negatives. It denotes how many of the relevant features were selected which is crucial when looking for the all relevant feature set. Precision is defined by TP / (TP+FP) with FP = false positives and describes what rate of false positives are part of the feature set. To get a balanced measure for a feature selection method, a combination of both is necessary. One can use the F_1 measure which is the harmonic mean of precision and recall:

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

The measures can be seen in Table II.

Before we evaluate the selection measures we confirm that all models had a proper fit. Listed in Table III are the training accuracies. One can see in the table that all classification models had accuracy values over 90% which signifies a sufficient fit of the data for all of them.

To evaluate the feature selection performance we mainly observe the F_1 score in Table II. Here our proposed *FRI* takes the lead overall with a near perfect score in all simulation sets. Depending on the presence of weakly relevant features, the other methods show loss of recall which leads to a reduced F_1 score. This is especially evident for *Sim2* and *Sim4* in the case

TABLE II
FEATURE SELECTION SCORE ON SIMULATED DATASETS. VALUES ARE SHOWING THE PERFORMANCE OF EACH METHOD TO CLASSIFY BETWEEN THE RELEVANCE OF INPUT FEATURES.

score	F1					precision					recall				
data	Sim1	Sim2	Sim3	Sim4	Sim5	Sim1	Sim2	Sim3	Sim4	Sim5	Sim1	Sim2	Sim3	Sim4	Sim5
Boruta	**0.98**	0.82	0.91	0.82	0.98	0.99	1.00	0.87	1.00	1.00	1.00	0.72	0.98	0.70	0.95
EFS	0.96	0.76	0.71	0.84	0.94	0.93	1.00	0.57	1.00	1.00	1.00	0.62	0.98	0.73	0.90
ElasticNet	0.62	0.84	0.44	0.82	0.80	0.46	0.74	0.28	0.69	0.67	1.00	0.98	1.00	1.00	1.00
FRI	**0.98**	**0.98**	0.99	**0.99**	**0.99**	0.98	1.00	0.98	0.99	1.00	0.99	0.97	1.00	0.98	0.99
StabilitySelection	0.77	0.75	**1.00**	0.91	0.27	1.00	1.00	1.00	1.00	1.00	0.62	0.60	1.00	0.83	0.16

TABLE III
AVERAGE TRAINING SET ACCURACY. IN THE CASE OF BORUTA THE INTERNAL RANDOMFOREST SCORE WAS REPORTED. FOR *EFS* ACCURACY IS NOT DEFINED.

	Boruta	EFS	ElasticNet	FRI	SS
Sim1	0.99	-	1.00	0.92	1.00
Sim2	0.97	-	1.00	0.96	1.00
Sim3	0.99	-	1.00	0.96	1.00
Sim4	0.97	-	1.00	0.93	1.00
Sim5	1.00	-	1.00	0.91	1.00
colp.	1.00	-	0.99	0.97	0.99
flip	1.00	-	0.90	0.82	0.90
spectf	1.00	-	0.99	0.92	0.98
t21	1.00	-	0.98	0.93	0.98
wbc	1.00	-	1.00	0.98	1.00

TABLE IV
ROC-AUC VALUES OF LOGISTIC REGRESSION MODEL USING FEATURES SELECTED BY LISTED MODELS. THE VALUES ARE AVERAGED OVER 50 BOOTSTRAPS.

	Boruta	EFS	EN	FRI	SS
colposcopy	0.568	0.586	0.640	**0.661**	0.625
flip	0.804	0.652	**0.815**	0.743	0.705
spectf	0.871	0.874	0.867	0.880	**0.888**
t21	0.971	0.977	0.971	0.975	**0.978**
wbc	0.997	0.998	0.998	0.998	0.999

of SS and EFS. The worst recall is achieved by SS for *Sim5* where it did not select many of the weakly relevant variables at all. SS still achieves slightly better scores in *Sim3* where no weakly relevant features are present.

2) Biomedical Data: The biomedical datasets are gathered from multiple studies and differ in size and type:

- *t21*: This set stems from a series of prenatal examinations of pregnant women. The goal of the examinations is the early diagnosis of chromosomal abnormalities, such as trisomy 21. The study covers sociodemographic, ultrasonographic and serum parameters which result in 18 usable features. The original set contains over 50.000 samples but only a low percentage ($\approx 0.8\%$) of abnormal samples. The data have been collected by the Fetal Medicine Centre at Kings College Hospital and University College London Hospital in London [32].
- *flip*: This set is used for the prediction of fibrosis. The diagnosis of fibrosis is represented as a score which is based on sociodemographic and serum parameters. The set consists of 118 patients and 19 features. The data were provided by the Department of Gastroenterology, Hepatology and Infectiology of the University Magdeburg [33].
- *spectf*: The *spectf* dataset consists of 44 features describing cardiac Single Proton Emission Computed Tomography (SPECT) images. Each of the 267 patients images were diagnosed as either normal or abnormal.
- *wbc*: The *wbc* dataset contains 32 markers for cell image based breast cancer diagnostics from 569 patients.
- *colposcopy*: Set with 69 Extracted structural features from videos acquired during colposcopies [34]. Classification of practitioners clinical judgment using the *Schiller* modality.

spectf, *wbc* and *colposcopy* were acquired through the UCI Machine Learning Repository [35].

The biomedical datasets are preprocessed before analysis. Samples with over 90% missing values are removed. Sets are split into stratified training and testing subsets. If samples still contain missing feature values, we replace them with the features training set mean in both subsets. Similarly, the z-score transformation is based on the training set and applied to both. In case the original set is imbalanced, we use the Synthetic Minority Over-sampling Technique (Smote) [36] in combination with a Nearest Neighbor cleaning rule [37] [38] as described in [39]. In one case (*t21*) with an extremely large majority class, we only perform downsampling. We perform the tests on 50 bootstrap replicates with sample size $0.7n$ with replacement.

To asses the quality of the feature selection on real-world datasets, we have to rely on the problem performance itself since no ground truth as regards feature relevance is available. We expect a FS method to pick features which contain information and a loss of features with information is signified in a decrease of performance. Instead of looking at each models internal accuracy score, we evaluate the selected feature sets by their discriminative power, whereby the latter is uniformly evaluated by a logistic regression model, which is a very popular model for predictive purposes in medical applications [40]. This model is trained using only the predicted feature set and hyperparameter optimization is performed to select the model with the best cross-validated regularization parameter. Finally, for this selected model the Receiver operation characteristics (ROC) on the holdout validation set are recorded. For the comparison, we look at the area under the curve (ROC-AUC), which is listed in Table IV. Here the AUC on the five datasets produces no clear overall superior method which is in

TABLE V
AVERAGE SELECTED FEATURE SET SIZE. ADDITIONALLY FOR *FRI* THE SIZE OF THE STRONGLY($_s$) AND WEAKLY ($_w$) RELEVANT FEATURE SET IS AVAILABLE.

	Boruta	EFS	EN	SS	FRI	FRI_s	FRI_w
Sim1	8.1	8.7	17.8	5.0	8.1	5.1	3.0
Sim2	14.3	12.3	26.6	12.1	19.4	12.4	7.0
Sim3	4.6	7.2	14.8	4.0	4.1	4.0	0.1
Sim4	12.6	13.2	26.2	15.0	17.9	17.9	0.0
Sim5	19.1	17.9	29.7	3.2	19.9	0.0	19.9
colp.	35.1	25.4	46.5	41.5	20.3	5.9	14.4
flip	18.8	8.1	16.9	9.1	8.9	8.8	0.1
spectf	44.0	20.3	43.1	5.9	19.9	5.9	14.0
t21	15.5	7.9	14.2	9.6	9.6	6.6	3.0
wbc	29.9	12.5	26.9	4.7	15.6	4.0	11.6

line with the expectation that the minimal optimal set is the objective of most methods and sufficient for prediction. On the *spectf*, *t21* and *wbc* datasets most methods produce very similar performing feature sets. In the case of *colposcopy* the feature set selected by FRI achieves the best performance. SS produces slighty better sets in two cases. The ElasticNet performs solid in all cases based on its very conservative selection method where informative features are not removed often.

This leads the part of the evaluation more concerned with the goal stated in the introduction. In the search for an interpretative and complete feature set we need to take the selected set size into account. Table V lists the average feature set sizes over all experiments. Because *FRI* provides additional information by not only conserving all weakly relevant features but also by denoting the feature class itself we can explicitly list those as well. As mentioned in the last paragraph, we can easily see that EN is very conservative in its selection. It produces by far the biggest feature sets with many false positives in the case of the *Sim* sets but also most likely in the real datasets. Similarly for Boruta, which achieves better precision in the generated data but still shows seemingly inflated set sizes. SS on the other hand exhibits very good precision overall. Interestingly the size of the sets chosen by *stability selection* is very similar to FRI_s, the set of strongly relevant features chosen by *FRI*. This indicates that *FRI* can find strongly relevant features with high precision, but also highlights the additional information provided by the weakly relevant features contained in FRI_w.

But why do we need the information in FRI_w?

B. Evaluation of Interactive Use

By having additional information available in the set of weakly relevant features FRI_w we can gain insights into the structure of the data. We can improve the design of models and diagnostic tests in biomedical applications. Our framework given in II-B allows introducing constraints into the model. This makes it possible to limit the contribution of certain features to specific intervals or a fixed value. These limits can come from prior knowledge of the practitioner and represent design goals or existing hypotheses. Depending on the chosen values the model and the resulting relevance bounds change and can be visualized again which lends itself to an iterative and interactive process. In the following, we are going to evaluate that use case on simulated data and on the *t21* data set.

The simulated set was generated according to III-A1. It consists of 8 features, 4 of which are strongly relevant, 3 of which are weakly relevant and one noise feature. Fig. 3 (a) shows the output of *FRI* without any constraints. The four strongly relevant features (1-4) are visible as four small rectangles with lower relevance bounds (the bottom part of the rectangle) bigger than zero. The model parameters allow some variation in their contribution to the model. The three weakly relevant features (5-7) are visible as three taller rectangles with equal height because they are perfectly correlated in the normalized space. They can replace each other in the model. This is apparent when we preset one of them (e.g feature 5) to the minimum and maximum relevance bound i.e. we calculate $\text{minRel}\mathbf{C}(\mathbb{D}, j, fc)$ and $\text{maxRel}\mathbf{C}(\mathbb{D}, j, fc)$ for all $j \neq 5$ where fc is $\mathbb{RI}_5^{min}$ or $\mathbb{RI}_5^{max}$.

In Fig. 3 (b) feature 5 was set to the minimum bound and the relevance bounds of other features are identical. That is because the other two features are still a correlated pair which allows the same degree of variability in contribution. When feature 5 is set to its maximum relevance bound in (c) we see that feature 6 and 7 have no contribution anymore. Additionally, all other relevance bounds are reduced to single values because the model in this state does not allow any more variability.

After this experiment, it is now interesting to apply this procedure on real data with functional associations between features. In Fig. 2 (a) the normal *FRI* output of the *t21* set is presented. As a reminder, this set consists of samples acquired in prenatal examinations of mothers and their unborn children. Features in the study included socioeconomic factors as well as ultrasound imaging metrics. Notably in the output of *FRI* are two weakly relevant features 1 and 6. Feature 1 represents the gestational age of the fetus in days ('GA-d') and feature 6 the crown rump length ('CRL') of the fetus, which is the length as indicated on an ultrasound machine. By intuition, we expect an association between the two measures. If we set one of the two features to its minimum relevance bound (Fig. 2 (b)) we see that feature 6 becomes strongly relevant in the model. This highlights the association between the two which is very useful in cases where it is not clear a priori. Furthermore, we can use this as a design tool to easily select 'better' features. If we find functional alternatives, we can exclude more expensive features in future experiments or tests.

C. Runtime

The computational runtime of a method is not only an indicator for its feasibility on bigger datasets but also especially important in the interactive use case where an analyst is actively involved in model refinement. In Fig. 4 we display aggregated mean runtime of the methods used in our evaluations on a single CPU thread. All the computations were limited to one thread, so an advantage of parallel processing is not taken

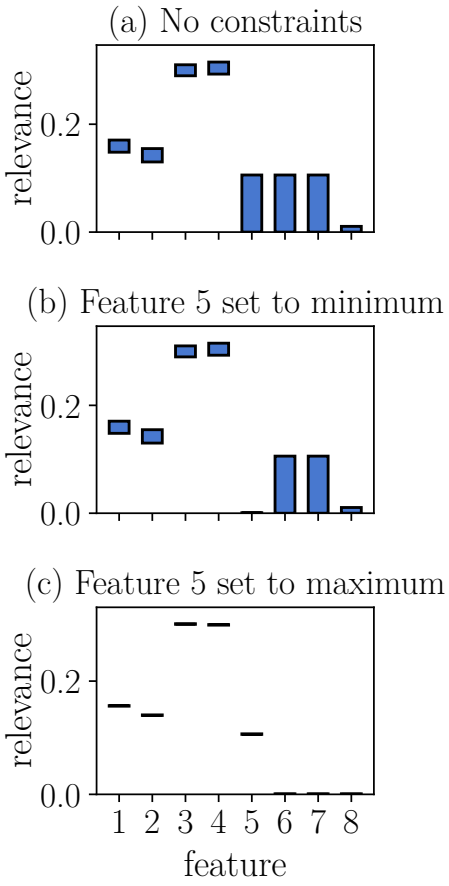

Fig. 3. Three subplots showing feature relevance bounds in different constraint situations according to section II-B. Classification data were simulated and consisted of 4 strongly relevant features (1-4), 3 weakly relevant (5-7) and one noise feature (8). Subplot (a) had no feature constraints. Subplot (b) shows the output when feature 5 is constrained to its minimum relevance value and (c) to its maximum value.

into account. EN performed best followed by SS. Both show steady runtimes over all types of data. *Borutas* runtime is very dependent on the density of the feature space and shows some variance in the case of *t21*. The runtime of *FRI* is similar to *Boruta* in most cases but takes a hit in smaller datasets because of the constant factor of sampling permutated features for feature classification. *EFS* shows the slowest performance in most cases, which clearly stems from its use of multiple complex underlying models at the same time.

Because relevance bounds can be solved independently in parallel, we provide the means to speed up computation by utilizing all available CPU cores on the machine. Additionally, we also tested running our program in conjunction with the distributed computation framework *Dask* which allows scaling up to any amount of separate computing nodes in a high performance cluster such as *Grid Engine* or even in cloud backends.

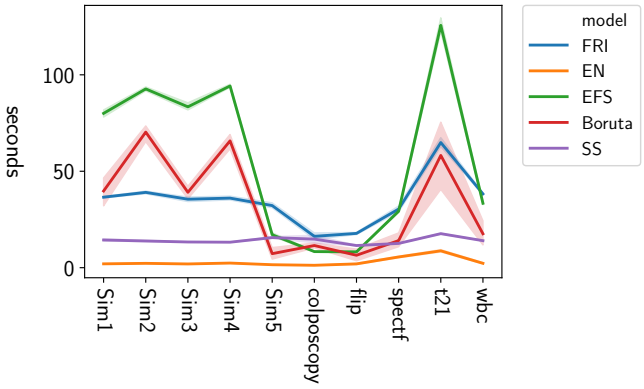

Fig. 4. Average runtime over all bootstraps with confidence intervals.

IV. Conclusion

We have presented the software library *FRI* to produce all relevant feature sets for general feature selection as well as perform interactive data exploration. We described how we implemented the algorithm from [25] and extended the method to allow a practitioner to include new constraints and experiment. We also proposed a threshold estimation method to reduce false positives which are common in all-relevant selection tasks.

In comparison with other methods, we showed that *FRI* can detect all relevant features in synthetic datasets while minimizing noise through its threshold estimation. On real datasets we showcased good selection performance and additional information provided by the weakly relevant feature set.

Our underlying method ensures to conserve all relevant variables while still maintaining interpretability. This is facilitated by the three relevance classes our method produces as well as the relevance bar representation which should enable better understanding for biological and medical experts in the future. In addition to facilitating understanding we also provide a way to incorporate prior knowledge to manipulate the model itself which should help in the design of new experiments and biomarkers for prediction models.

Acknowledgments

The authors would like to thank Professor Kypros Nicolaides and Dr. Argyro Syngelaki from Fetal Medicine Foundation for making available the *t21* data used in this study. We are also grateful for the *flip* datasets provided by Professor Ali Canbay, Department of Gastroenterology, Hepatology, and Infectiology of the University Hospital Magdeburg.

References

[1] Kononenko I. Machine Learning for Medical Diagnosis: History, State of the Art and Perspective. Artificial Intelligence in Medicine. 2001 Aug;23(1):89–109.

[2] Bellazzi R, Zupan B. Predictive Data Mining in Clinical Medicine: Current Issues and Guidelines. International Journal of Medical Informatics. 2008 Feb;77(2):81–97.

[3] Cruz JA, Wishart DS. Applications of Machine Learning in Cancer Prediction and Prognosis. Cancer Informatics. 2006 Jan;2:117693510600200030.

[4] Vellido Alcacena A, Guerrero M, D J, Lisboa PJG. Making Machine Learning Models Interpretable. In: ESANN 2012 Proceedings: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning: Bruges (Belgium), 25-27 April, 2012; 2012. p. 163–172.

[5] Guyon I, Elisseeff A. An Introduction to Variable and Feature Selection. Journal of Machine Learning Research. 2003;3(Mar):1157–1182.

[6] Yu L, Liu H. Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. In: Proceedings of the Twentieth International Conference on International Conference on Machine Learning. ICML'03. AAAI Press; 2003. p. 856–863.

[7] Kohavi R, John GH. Wrappers for Feature Subset Selection. Artif Intell. 1997 Dec;97(1-2):273–324.

[8] Tibshirani R. Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society Series B (Methodological). 1996;58(1):267–288.

[9] Applications of Support Vector Machine (SVM) Learning in Cancer Genomics. Cancer Genomics & Proteomics. 2018 Jan;15(1).

[10] Zhao P, Yu B. On Model Selection Consistency of Lasso. Journal of Machine Learning Research. 2006;7:2541–2563.

[11] Greenshtein E, Ritov Y. Persistence in High-Dimensional Linear Predictor Selection and the Virtue of Overparametrization. Bernoulli. 2004 Dec;10(6):971–988.

[12] Bell DA. A Formalism for Relevance and Its Application in Feature Subset Selection;p. 21.

[13] van Dam S, Võsa U, van der Graaf A, Franke L, Magalhães D, Pedro Ja. Gene Co-Expression Analysis for Functional Classification and Gene–Disease Predictions. Briefings in Bioinformatics;.

[14] Toloşi L, Lengauer T. Classification with Correlated Features: Unreliability of Feature Ranking and Solutions. Bioinformatics. 2011 Jul;27(14):1986–1994.

[15] Kumar V. Feature Selection: A Literature Review. The Smart Computing Review. 2014 Jun;4(3).

[16] Zou H, Hastie T. Regularization and Variable Selection via the Elastic Net. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2005;67(2):301–320.

[17] LeCun Y, Jackel L, Bottou L, Cortes C, Denker JS, Drucker H, et al. Learning Algorithms for Classification: A Comparison on Handwritten Digit Recognition. Neural networks: the statistical mechanics perspective. 1995;261:276.

[18] Lagani V, Athineou G, Farcomeni A, Tsagris M, Tsamardinos I. Feature Selection with the R Package MXM: Discovering Statistically-Equivalent Feature Subsets. ArXiv e-prints. 2016 Nov;1611:arXiv:1611.03227.

[19] Meinshausen N, Bühlmann P. Stability Selection. Journal of the Royal Statistical Society Series B, Statistical methodology. 2010;.

[20] Shah RD, Samworth RJ. Variable Selection with Error Control: Another Look at Stability Selection: *Another Look at Stability Selection*. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2013 Jan;75(1):55–80.

[21] Kursa MB, Rudnicki WR, et al. Feature Selection with the Boruta Package. J Stat Softw. 2010;36(11):1–13.

[22] Degenhardt F, Seifert S, Szymczak S. Evaluation of Variable Selection Methods for Random Forests and Omics Data Sets. Briefings in Bioinformatics. 2017 Oct;.

[23] Neumann U, Genze N, Heider D. EFS: An Ensemble Feature Selection Tool Implemented as R-Package and Web-Application. BioData Mining. 2017 Jun;10:21.

[24] Neumann U, Riemenschneider M, Sowa JP, Baars T, Kälsch J, Canbay A, et al. Compensation of Feature Selection Biases Accompanied with Improved Predictive Performance for Binary Classification by Using a Novel Ensemble Feature Selection Approach. BioData Mining. 2016 Nov;9:36.

[25] Göpfert C, Pfannschmidt L, Göpfert JP, Hammer B. Interpretation of Linear Classifiers by Means of Feature Relevance Bounds. Neurocomputing. 2018 Jul;298:69–79.

[26] Pfannschmidt L, Jakob J, Biehl M, Tino P, Hammer B. Feature Relevance Bounds for Ordinal Regression. In: Proceedings of the 27th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Michel Verleysen, editor; 2019. Accepted.

[27] Chang YW, Lin CJ. Feature Ranking Using Linear SVM. In: Causation and Prediction Challenge; 2008. p. 53–64.

[28] Yao L, Zeng F, Li DH, Chen ZG. Sparse Support Vector Machine with Lp Penalty for Feature Selection. Journal of Computer Science and Technology. 2017 Jan;32(1):68–77.

[29] Anguita D, Boni A, Ridella S. Evaluating the Generalization Ability of Support Vector Machines through the Bootstrap. Neural Processing Letters. 2000 Feb;11(1):51–58.

[30] Geisser S. Predictive Inference. CRC Press; 1993.

[31] Kursa MB, Rudnicki WR. The All Relevant Feature Selection Using Random Forest. CoRR. 2011;abs/1106.5112.

[32] Nicolaides KH, Spencer K, Avgidou K, Faiola S, Falcon O. Multicenter Study of First-Trimester Screening for Trisomy 21 in 75 821 Pregnancies: Results and Estimation of the Potential Impact of Individual Risk-Orientated Two-Stage First-Trimester Screening: First-Trimester Screening for Trisomy 21. Ultrasound in Obstetrics and Gynecology. 2005 Mar;25(3):221–226.

[33] Sowa JP, Heider D, Bechmann LP, Gerken G, Hoffmann D, Canbay A. Novel Algorithm for Non-Invasive Assessment of Fibrosis in NAFLD. PLOS ONE. 2013 Apr;8(4):e62439.

[34] Fernandes K, Cardoso JS, Fernandes J. Transfer Learning with Partial Observability Applied to Cervical Cancer Screening. In: IbPRIA; 2017. .

[35] Dheeru D, Karra Taniskidou E. UCI Machine Learning Repository. 2017;.

[36] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic Minority Over-Sampling Technique. Journal of Artificial Intelligence Research. 2002 Jun;16:321–357.

[37] Wilson DL. Asymptotic Properties of Nearest Neighbor Rules Using Edited Data. IEEE Transactions on Systems, Man, and Cybernetics. 1972 Jul;SMC-2(3):408–421.

[38] Laurikkala J. Improving Identification of Difficult Small Classes by Balancing Class Distribution. In: Goos G, Hartmanis J, van Leeuwen J, Quaglini S, Barahona P, Andreassen S, editors. Artificial Intelligence in Medicine. vol. 2101. Springer Berlin Heidelberg; 2001. p. 63–66.

[39] Batista GEAPA, Prati RC, Monard MC. A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. SIGKDD Explor Newsl. 2004 Jun;6(1):20–29.

[40] Bagley SC, White H, Golomb BA. Logistic Regression in the Medical Literature: Standards for Use and Reporting, with Particular Attention to One Medical Domain. Journal of Clinical Epidemiology. 2001 Oct;54(10):979–985.