BioData Mining

SOFTWARE ARTICLE

Open Access

EFS: an ensemble feature selection tool implemented as R-package and web-application

Ursula Neumann[1,2,3], Nikita Genze[1] and Dominik Heider[1,2,3*]

*Correspondence:
d.heider@wz-straubing.de
[1]Straubing Center of Science, Schulgasse 22, 94315 Straubing, Germany
[3]Wissenschaftszentrum Weihenstephan, Technische Universität München, 85354 Freising, Germany
Full list of author information is available at the end of the article

Abstract

Background: Feature selection methods aim at identifying a subset of features that improve the prediction performance of subsequent classification models and thereby also simplify their interpretability. Preceding studies demonstrated that single feature selection methods can have specific biases, whereas an ensemble feature selection has the advantage to alleviate and compensate for these biases.

Results: The software EFS (Ensemble Feature Selection) makes use of multiple feature selection methods and combines their normalized outputs to a quantitative ensemble importance. Currently, eight different feature selection methods have been integrated in EFS, which can be used separately or combined in an ensemble.

Conclusion: EFS identifies relevant features while compensating specific biases of single methods due to an ensemble approach. Thereby, EFS can improve the prediction accuracy and interpretability in subsequent binary classification models.

Availability: EFS can be downloaded as an R-package from CRAN or used via a web application at http://EFS.heiderlab.de.

Keywords: Machine learning, Feature selection, Ensemble learning, R-package

Background

In the field of data mining, feature selection (FS) has become a frequently applied preprocessing step for supervised learning algorithms, thus a great variety of FS techniques already exists. They are used for reducing the dimensionality of data by ranking features in order of their importance. These orders can then be used to eliminate those features that are less relevant to the problem at hand. This improves the overall performance of the model because it addresses the problem of overfitting. But there are several reasons that can cause instability and unreliability of the feature selection, e.g., the complexity of multiple relevant features, a small-n-large-p-problem, such as in high-dimensional data [1, 2], or when the algorithm simply ignores stability [3, 4]. In former studies, it has been demonstrated that a single optimal FS method cannot be obtained [5]. For example, the Gini-coefficient is widely used in predictive medicine [6, 7], but it has also been demonstrated to deliver unstable results in unbalanced datasets [8, 9]. To counteract instability and therewith unreliability of feature selection methods, we developed an FS procedure for binary classification, which can be used, e.g., for random clinical trials. Our new approach ensemble feature selection (EFS) [10] is based on the idea of ensemble learning [11, 12], and thus is based on the aggregation of multiple FS methods. Thereby a quantification of the importance scores of features can be obtained and the method-specific biases can be compensated. In the current paper we introduce an R-package and a web server based on the EFS method. The user of the R-package as well as the web application can decide which FS methods should be conducted. Therewith, the web server and the R-package can be applied to perform an ensemble of FS methods or to calculate an individual FS score.

Implementation

We used existing implementations in **R** (http://www.r-project.org/) for our package EFS. The following section will briefly introduce our methodology. For deeper insights please refer to [10]. Our EFS currently incorporates eight feature selection methods for binary classifications, namely median, Pearson- and Spearman-correlation, logistic regression, and four variable importance measures embedded in two different implementations of the random forest algorithm, namely *cforest* [9] and *randomForest* [13].

Median

This method compares the positive samples (class = 1) with negative samples (class = 0) by a Mann-Whitney-U Test. The resulting p-values are used as a measure of feature importance. Thus, a smaller p-value indicates a higher importance.

Correlation

We used the idea of a fast correlation based filter of of Yu and Liu [14] to select features that are highly correlated with the dependent variable, but show only low correlation with other features. The fast correlation based filter eliminates features with high correlation with other features to avoid multicollinearity. The eliminated features get an importance value of zero. Two correlation coefficients, namely the Pearson product-moment and the Spearman rank correlation coefficient were adopted and their p-values were used as importance measure.

Logistic regression

The weighting system (i.e., β-coefficients) of the logistic regression (LR) is another popular feature selection method. As preprocessing step a Z-transformation is conducted to ensure comparability between the different ranges of feature values. The β-coefficients of the resulting regression equation represent the importance measure.

Random forest

Random forests (RFs) are ensembles of multiple decision trees, which gain their randomness from the randomly chosen starting feature for each tree. There are different implementations of the RF algorithm in R available, which offer diverse feature selection methods. On the one hand we incorporated the *randomForest* implementation based on the classification and regression tree (CART) algorithm by Breiman [13]. The *cforest* implementation from the party package, on the other hand, uses conditional trees for the purpose of classification and regression (cf. [15]). In both implementations an error-rate-based importance measure exists. The error-rate-based methods measure the difference before and after permuting the class variable. Due to their dependency on the underlying trees, results are varying for both error-rates. The *randomForest* approach also provides an importance measure based on the Gini-index, which measures the node impurity in the trees. Whereas in *cforest* an AUC-based variable importance measure is implemented. The AUC (area under the curve) is the integral of the receiver operating characteristics (ROC) curve. The AUC-based variable importance measure works to the error-rate-based one, but instead of computing the error rate for each tree before and after permuting a feature, the AUC is computed.

Ensemble learning

The results of each individual FS methods are normalized to a common scale, an interval from 0 to $\frac{1}{n}$, where n is the number of conducted FS methods chosen by the user. Thereby we ensure the comparability of all FS methods and conserve the distances between the importance of one feature to another.

R-package

The EFS package is included in the Comprehensive R Archive Network (CRAN) and can be directly downloaded and installed by using the following R command:

```
install.packages("EFS")
```

In the following, we introduce EFS's three functions `ensemble_fs`, `barplot_fs` and `efs_eval`. A summary of all commands and parameters is shown in Table 1.

Table 1 Method overview

Command	Parameters	Information
ensemble_fs	data	object of class data.frame
	classnumber	index of variable for binary classification
	NA_threshold	threshold for deletion of features with a greater proportion of NAs
	cor_threshold	correlation threshold within features
	runs	amount of runs for randomForest and cforest
	selection	selection of feature selection methods to be conducted
barplot_fs	name	character string giving the name of the file
	efs_table	table object of class matrix retrieved from ensemble_fs
efs_eval	data	object of class data.frame
	efs_table	table object of class matrix retrieved from ensemble_fs
	file_name	character string, name which is used for the two possible PDF files.
	classnumber	index of variable for binary classification
	NA_threshold	threshold for deletion of features with a greater proportion of NAs
	logreg	logical value indicating whether to conduct an evaluation via logistic regression or not
	permutation	logical value indicating whether to conduct a permutation of the class variable or not
	p_num	number of permutations; default set to a 100
	variances	logical value indicating whether to calculate the variances of importances retrieved from bootstrapping or not
	jaccard	logical value indicating whether to calculate the Jaccard-index or not
	bs_num	number of bootstrap permutations of the importances
	bs_percentage	proportion of randomly selected samples for bootstrapping

The R-package EFS provides three functions

`ensemble_fs`

The main function is `ensemble_fs`. It computes all FS methods which are chosen via the `selection` parameter and gives back a table with all normalized FS scores in a range between 0 and $\frac{1}{n}$, where n is the number of incorporated feature selection methods. Irrelevant features (e.g., those with too many missing values) can be deleted.

```
ensemble_fs(data, classnumber,
            NA_threshold, cor_threshold,
            runs, selection)
```

The parameter `data` is an object of class `data.frame`. It consists of all features and the class variables as columns. The user has to set the parameter `classnumber`, which represents the column number of the class variable, i.e., the dependent variable for classification. `NA_threshold` represents a threshold concerning the allowed proportion of missing values (NAs) in a feature column. The default value is set to 0.2, meaning that features with more than 20% of NAs are neglected by the EFS algorithm. The `cor_threshold` parameter is only relevant for the correlation based filter methods. It determines the threshold of feature-to-feature correlations [14]. The default value of `cor_threshold` is 0.7. The results of RF-based FS methods vary due to the randomness of their underlying algorithms. To obtain reliable results, the RF methods are conducted several times and averaged over the number of runs. This parameter, namely `runs`, is set to 100 by default. The user can select the FS methods for the EFS approach by using the `selection` parameter. Due to the high computational costs of the RFs, the default selection is set to

```
selection = c(TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE),
```

meaning that the two FS methods of the conditional random forest are not used by default.

`barblot_fs`

The `barblot_fs` function sums up all individual FS scores based on the output of `ensemble_fs` and visualizes them in an cumulative barplot.

```
# Create a cumulative barplot based on the output from EFS
barplot_fs(name, efs_table)
```

The `barplot_fs` function uses the output of the `ensemble_fs` function, namely the `efs_table`, as input. The parameter `name` represents the filename of the resulting PDF, which is saved in the current working directory.

`efs_eval`

The `efs_eval` function provides several tests to evaluate the performance and validity of the EFS method. The parameters `data, efs_table, file_name, classnumber` and `NA_threshold` are identical to the corresponding parameters in the `ensemble_fs` function:

```
efs_eval(data, efs_table, file_name,
         classnumber , NA_threshold,
         logreg = TRUE,
         permutation = TRUE, p_num,
         variances = TRUE, jaccard = TRUE,
         bs_num, bs_percentage).
```

Performance evaluation by logistic regression

The performance of the EFS method can automatically be evaluated based on a logistic regression (LR) model, by setting the parameter `logreg = TRUE`. `efs_eval` uses an LR model of the selected features with a leave-one-out cross-validation (LOOCV) scheme, and additionally trains an LR model with all available feature in order to compare the two LR models based on their ROC curves and AUC values with ROCR [16] and pROC based on the method of DeLong et al. [17]. A PDF with the ROC curves is automatically saved in the working directory.

Permutation of class variable

In order to estimate the robustness of the resulting LR model, permutation tests [18, 19] can be automatically performed, by setting the parameter `permutation = TRUE`. The class variable is randomly permuted `p_num` times and logistic regression is conducted. The resulting AUC values are then compared with the AUC from the original LR model using a Student's t-Test. By default, `p_num` is set to 100 permutations.

Variance of feature importances

If the parameter `variances` is `TRUE` an evaluation of the stability of feature importances will be conducted by a bootstrapping algorithm. The samples are permuted for `bs_num` times and a subset of the samples (`bs_percentage`) is chosen to calculate the resulting feature importances. By default, the function chooses 90% of the samples and uses 100 repetitions. Finally, the variances of the importance values are reported.

Jaccard-index

The Jaccard-index measures the similarity of the feature subsets selected by permuted EFS iterations:

$$J(S_1, \ldots, S_n) = \frac{|S_1 \cap \ldots \cap S_n|}{|S_1 \cup \ldots \cup S_n|},$$

where S_i is the subset of features at the i-th iteration, for $i = 1, \ldots, n$. The value of the Jaccard-index varies from 0 to 1, where 1 implies absolute similarity of subsets. If `jaccard = TRUE` is set, the Jaccard-index of the subsets retrieved from the bootstrapping algorithm is calculated.

Availability and requirements

The package is available for R-users under the following requirements:

- **Project name:** Ensemble Feature Selection
- **Project home page (CRAN):** http://cran.r-project.org/web/packages/EFS
- **Operating system (s):** Platform independent
- **Programming language:** R ($\geq$ 3.0.2)
- **License:** GPL ($\geq$ 2)
- **Any restrictions to use by non-academics:** none

Due to the high relevance of our EFS tool for researchers who are not very familiar with R (e.g., medical practitioners), we also provide a web application at http://EFS.heiderlab.de. It contains the functions `ensemble_fs` and `barplot_fs`. Therefore no background knowledge in R is necessary to use our new EFS software.

Results

The dataset SPECTF has been obtained from the UCI Machine Learning Repository [20] and is used as an example. It describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images. The class-variable represents normal (= 0) and abnormal (= 1) results and can be found in the first column of the table of the file SPECTF.csv at the UCI repository. In general, the EFS approach accepts all types of data, i.e., all types of variables, except categorical variables. These variables have to be transformed to dummy variables in advance. Data has to be combined in a single file with one column indicating the class variable with 1 and 0, e.g., representing patients and control samples, or, e.g., positive and negative samples. After loading the dataset, we compute the EFS and store it in the variable "efs":

```
library(EFS)
# Loading dataset in environment
efsdata <- read.table("SPECTF.csv", sep = ";")
# Start feature selection
efs <- ensemble_fs(data = efsdata, classnumber = 1,
                   NA_threshold = 0.2, cor_threshold = 0.7,
                   runs = 100, selection = rep(TRUE, 8))
```

The results can be visualized by the `barplot_fs` function:

```
# Create a cumulative barplot based on the output from efs
barplot_fs("SPECTF", efs)
```

The output is a PDF named "SPECTF.pdf". Figure 1 shows this cumulative barplot, where each FS method is given in a different color. Various methods to evaluate the stability and reliability of the EFS results are conducted by the following command:

```
# Create a ROC Curve based on the output from efs
eval_tests <- efs_eval(data = efs_data, efs_table = efs,
                       file_name = "SPECTF",
                       classnumber = 1, NA_threshold = 0.2,
                       logreg = TRUE,
                       permutation = TRUE, p_num = 100,
                       variances = TRUE, jaccard = TRUE,
                       bs_num = 100, bs_percentage = 0.9)
```

The user retrieves two PDF files. Firstly, the resulting ROC curves of the LR test ("SPECTF_ROC.pdf") including the p-value, according to Fig. 2. The p-value clearly shows that there is a significant improvement in terms of AUC of the LR with features selected by the EFS method compared the LR model without feature selection. Additionally, Fig. 3 shows the file "SPECTF_Variances.pdf", in which boxplots of the importances retrieved from the bootstrapping approach are given. The calculated variances can be accessed in the eval_tests output. A low variance implies that the importance of a feature is stable and reliable.

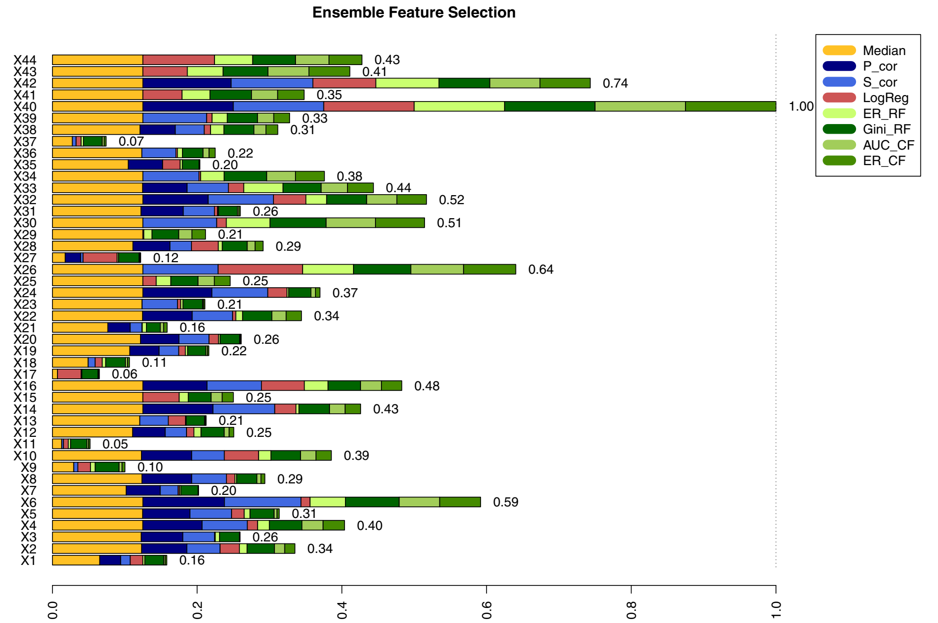

Fig. 1 Cumulative barplot retrieved from `barplot_fs` function of R-package EFS

An additional example is provided in the documentation of the R-package on a dataset consisting of weather data from the meteorological stations in Frankfurt(Oder), Germany in February 2016.

Conclusion

The EFS R-package and the web-application are implementations of an ensemble feature selection method for binary classifications. We showed that this method can improve the prediction accuracy and simplifies the interpretability by feature reduction.

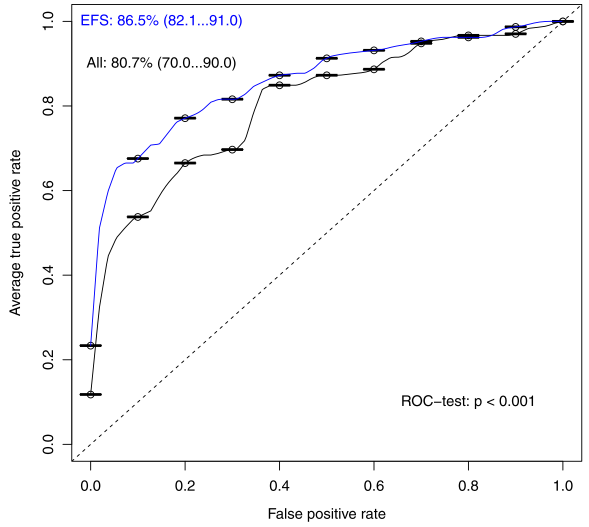

Fig. 2 Performance of LR model. On the *y-axis* the average true positive rate (i.e., sensitivity) and on the *x-axis* the false positive rate (i.e., 1-specificity) is shown. Two ROC curves are shown: of all features (*black*) and the EFS selected features (*blue*). The *dotted line marks* the performance of random guessing

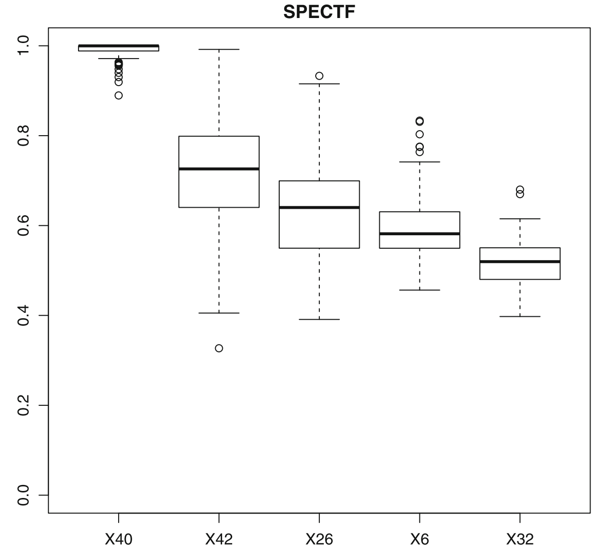

Fig. 3 Boxplot of importances retrieved from the bootstrapping algorithm

Abbreviations

AUC: Area under the curve; CART: Classification and regression tree; CRAN: Comprehensive R archive network; EFS: Ensemble feature selection; FS: Feature selection; LR: logistic regression; LOOCV: leave-one-out cross validation; RF: random forest; ROC: receiver operating characteristic

Acknowledgments

We are grateful to the UCI Machine Learning Repository for granting access to a great variety of datasets.

Funding

This work was supported by the German Research Foundation (DFG) and the Technische Universität München within the funding program Open Access Publishing and the Deichmann Foundation, which had no role in study design, collection, analysis, and interpretation of data, and in writing the manuscript.

Availability of data and materials

The dataset *SPECTF* in this article is available in the UCI Machines Learning repository, http://archive.ics.uci.edu/ml.

Authors' contributions

UN and NG have implemented the R-package. UN has implemented the web application and drafted the manuscript. DH designed and supervised the study. DH revised the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Consent for publication

Not applicable.

Ethics approval and consent to participate

Not applicable.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

[1]Straubing Center of Science, Schulgasse 22, 94315 Straubing, Germany. [2]University of Applied Science, Weihenstephan-Triesdorf, 85354 Freising, Germany. [3]Wissenschaftszentrum Weihenstephan, Technische Universität München, 85354 Freising, Germany.

Received: 23 November 2016 Accepted: 12 June 2017
Published online: 27 June 2017

References

1. Dybowski JN, Heider D, Hoffmann D. Structure of hiv-1 quasi-species as early indicator for switches of co-receptor tropism. AIDS Res Ther. 2010;7:41.
2. Pyka M, Hahn T, Heider D, Krug A, Sommer J, Kircher T, Jansen A. Baseline activity predicts working memory load of preceding task condition. Hum Brain Mapp. 2013;34(11):3010–22.
3. Jain A, Zongker D. Feature selection: evaluation, application, and small sample performance. IEEE Trans Pattern Anal Mach Intell. 1997;19(2):153–8.
4. He Z, Yu W. Stable feature selection for biomarker discovery. Comput Biol Chem. 2010;34:215–25.
5. Yang YHY, Xiao Y, Segal MR. Identifying differentially expressed genes from microarray experiments via statistic synthesis. Bioinformatics. 2004;21(7):1084–93.
6. Leclerc A, Lert F, Cecile F. Differential mortality: Some comparisons between england and wales, finland and france, based on inequality measures. Int J Epidemiol. 1990;19(4):1001–10.
7. Llorca J, Delgado-Rodríguez M. Visualising exposure-disease association: the lorenz curve and the gini index. Med Sci Monit. 2002;8(10):193–7.
8. Sandri M, Zuccolotto P. A bias correction algorithm for the gini variable importance measure in classification trees. J Comput Graph Stat. 2008;17(3):611–28.
9. Boulesteix AL, Janitza S, Kruppa J, König IR. Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. Wiley Interdiscip Rev Data Min Knowl Discov. 2012;2(6): 493–507.
10. Neumann U, Riemenschneider M, Sowa JP, Baars T, Kälsch J, Canbay A, Heider D. Compensation of feature selection biases accompanied with improved predictive performance for binary classification by using a novel ensemble feature selection. BioData Min. 2016;9(1):36.
11. Saeys Y, Abeel T, Van de Peer Y. Robust feature selection using ensemble feature selection techniques. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Berlin: Springer-Verlag; 2008. p. 313–25.
12. Abeel T, Helleputte T, Van de Peer Y, Dupont P, Saeys Y. Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. Bioinformatics. 2010;26(3):392–8.
13. Breiman L. Random forests. Mach Learn. 2001;45(1):5–32.
14. Yu L, Liu H. Efficient feature selection via analysis of relevance and redundancy. J Mach Learn Res. 2004;5:1205–24.
15. Strobl C, Boulesteix AL, Kneib T, Augustin T, Zeileis A. Conditional variable importance for random forest. BMC Bioinforma. 2006;9(307):1–11.
16. Sing T, Sander O, Beerenwinkel N, Lengauer T. Rocr: visualizing classifier performance in r. Bioinformatics. 2005;21(20):3940–41.
17. DeLong ER, DeLong DM, Clarke-Pearson DL. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. Biometrics. 1988;44(3):837–45.
18. Barbosa E, Röttger R, Hauschild AC, Azevedo V, Baumbach J. On the limits of computational functional genomics for bacterial lifestyle prediction. Brief Funct Genomics. 2014;13:398–408.
19. Sowa JP, Atmaca Ö, Kahraman A, Schlattjan M, Lindner M, Sydor S, Scherbaum N, Lackner K, Gerken G, Heider D, Arteel GE, Erim Y, Canbay A. Non-invasive separation of alcoholic and non-alcoholic liver disease with predictive modeling. PLOS ONE. 2014;9(7):101444.
20. Lichman M. UCI Machine Learning Repository. 2013. http://archive.ics.uci.edu/ml.