Transactions on Machine Learning and Artificial Intelligence - Vol. 9, No. 4
Publication Date: August, 25, 2021
DOI:10.14738/tmlai.94.10101.
Noureldien, N. A, & Mahmoud, S. (2021). The Efficiency of Aggregation Methods in Ensemble Filter Feature Selection Models. Transactions on Machine Learning and Artificial Intelligence, 9(4). 39-51.

The Efficiency of Aggregation Methods in Ensemble Filter Feature Selection Models

Noureldien A. Noureldien
Department of Computer Science, University of Science and Technology
Omdurman, Sudan

Safaa Mahmoud
Department of Computer Science, University of Science and Technology
Omdurman, Sudan

ABSTRACT

Ensemble feature selection is recommended as it proves to produce a more stable subset of features and a better classification accuracy when compared to the individual feature selection methods. In this approach, the output of feature selection methods, called base selectors, are combined using some aggregation methods. For filter feature selection methods, a list aggregation method is needed to aggregate output ranked lists into a single list, and since many list aggregation methods have been proposed the decision on which method to use to build the optimum ensemble model is a de facto question. In this paper, we investigate the efficiency of four aggregation methods, namely; Min, Median, Arithmetic Mean, and Geometric Mean. The performance of aggregation methods is evaluated using five datasets from different scientific fields with a variant number of instances and features. Besides, the classifies used in the evaluation are selected from three different classes, Trees, Rules, and Bayes. The experimental results show that 11 out of the 15 best performance results are corresponding to ensemble models. And out of the 11 best performance ensemble models, the most efficient aggregation methods are Median (5/11), followed by Arithmetic Mean (3/11) and Min (3/11). Also, results show that as the number of features increased, the efficient aggregation method changes from Min to Median to Arithmetic Mean. This may suggest that for a very high number of features the efficient aggregation method is the Arithmetic Mean. And generally, there is no aggregation method that is the best for all cases.

Keywords: Filter feature selection methods, Ensemble models, Aggregation methods, Ranked lists.

INTRODUCTION

Feature selection is a well-known preprocessing method in data mining. Feature selection is targeting reducing the dimensionality of the dataset while attaining the basic characteristics of the original data set. The dimensionality reduction improves the speed, performance, and interpretability of the data mining model [1]. Thus, the process of feature selection aims to select a small subset of relevant features while still able to construct learning models with optimal predictive performance [2].

Feature selection methods are classified into three categories; filters, wrappers, and embedded methods [3][4]. To obtain a representative subset of features out of the original set of dataset features, there are two different approaches to evaluate the dataset features, the first approach is called individual evaluation. In this approach, each feature is evaluated individually by the selection method, and a ranking of features is returned by assigning a score or weight to each feature. In the second approach, called subset evaluation, a subset of features is selected by the selection method and evaluated according to an optimality criterion until a final subset of selected features is obtained. In this work, we focus only on filter feature selection methods whose output is a ranked list of features rather than a subset of features.

Ensemble techniques are primarily applied in machine learning to improve classification and prediction tasks. In this technique, the output of multiple classifiers is combined to produce a final output. Recently the same techniques are applied in the feature selection process to improve the stability of the feature selection process [5] [6][7]. The main focus of the ensemble feature ranking approach is on improving classification performance through the combination of outputs ranks of feature selection.

There are two approaches are used to construct an ensemble of feature selection; either multiple feature selection methods are used with a single dataset and their output is combined or a single feature selection method is used with different variants of the dataset and the outputs are combined. In the first approach, the feature selection methods are called base selectors, and the ensemble is said to be heterogeneous due to the different types of selectors, while the second is called homogenous since a single elector is used [8][9].

In this paper, we deal with filter heterogeneous ensemble feature selection models, i.e. ensemble models where base selectors are different filter methods. Filter methods are known to produce a ranked list of features rather than a subset of the original features. Thus, to construct an ensemble model for such methods a list aggregation method is needed to aggregate ranked lists into a single list. And as many aggregation methods have been proposed to aggregate lists, a question about which aggregation method can be more efficient in the construction of an ensemble model is a de facto question.

Many aggregation methods have been proposed in the literature to aggregate multiple ranked lists into a final list. Examples include Mean, Median, Highest Rank, Lowest Rank, Round Robin, Stability Selection, Exponential Weighting, and Robust Rank Aggregation. All of these aggregation methods assume that the ranked lists being combined assign a value to each feature, from 1 to N (for N features), where the best feature is assigned number 1, the second-best feature is 2, and so on until the worst feature is assigned N.

To provide an answer to the aforementioned question, aggregation methods efficiency should be examined using a wide variety of values for variables that affect the efficiency of the constructed ensemble models, namely; the number of filters (base selectors), the number of datasets and their characteristics, and the number and types of the evaluator classifiers.

In this work, we investigate the performance of four simple aggregation methods, namely; min, median, arithmetic mean, and geometric mean. To generalize results, we use five datasets from

different fields with a variant number of instances and features. Also, the classifies used to evaluate the performance are selected from three different classes; Trees, Rules, and Bayes.

The remainder of this paper is organized as follows. Section 2 explores the related works. Section 3 is material and methods. Section4 shows experimental results. Lastly, Section 5 presents our conclusions and recommendations for future work.

RELATED WORK

Few kinds of research have studied or compare aggregation methods (or rank aggregation techniques) that are used in forming ensemble feature selection models. One of the studies that attempt to compare rank aggregation methods is presented in [10]. In this study, nine rank aggregation techniques (Mean, Median, Highest Rank, Lowest Rank, Stability Selection, Exponential Weighting, Enhanced Borda, Round Robin, and Robust Rank aggregation), ten different feature selection methods (chi-squared, information gain, ReliefF, Mutual Information, Kolmogorov-Smirnov statistic, Deviance, Geometric Mean, Area Under the ROC Curve, Area Under the Precision-Recall Curve, and Signal-to-Noise) are compared across twenty-six bioinformatics datasets. Their results show that certain aggregation techniques are very similar to each other, while others are unique in that they are not similar to the other techniques. Additionally, it was found that as the size of the feature subset increases, the similarity between the techniques increases.

An in-depth comparison between two aggregation techniques is given in [11] namely, the Mean and Robust rank aggregation technique which is designed specifically for bioinformatics. The experimental results show that in general mean aggregation will outperform Robust Rank Aggregation in terms of classification performance while being significantly simpler to implement and perform.

A comprehensive study that evaluates nine different rank aggregation techniques is given in [12]. The aim is to find the best aggregation method for classification models to be used with gene microarray data for distinguishing between cancerous and non-cancerous cells.

Nine aggregation methods; Mean, Median, Highest Rank, Lowest Rank, Stability Selection, Exponential Weighting, Enhanced Borda, Round Robin, and Robust Rank Aggregation are tested using a homogeneous ensemble model with twenty-five base selectors. Each ensemble model for a base selector is built out of fifty perpetration of the dataset. To test the performance of the constructed ensemble models, eleven bioinformatics datasets and five learners are used. Their results show that the aggregation method "Lowest Rank" is the worst aggregation technique, while other techniques perform almost similarly. They conclude that a simple technique such as Mean aggregation is preferable due to its low computation time.

The above previous studies have a limited scope, either in terms of datasets field (for example bioinformatics), or in terms of classifiers used to test the aggregation techniques (for example classifiers from one class such as Decision Tree class).

In this study, we attempt to make the results as general as possible. To do this, the performance of the four aggregation methods is evaluated using five datasets from different scientific fields

with a variant number of instances and features. Also, the classifiers used to evaluate the performance are selected from three different classes (Trees, Rules, Bayes).

MATERIAL AND METHODS

This section illustrates the various materials and methods that we use in our practical experimentation of the efficiency of the standalone filter base selectors and the constructed ensemble models which are build using the aggregation methods under investigation.

Analysis tool and Computational Environment

In practical experiments we use a Toshiba laptop that has the system model C50-B, with Intel(R) Celeron(R) CPU N2830 @ 2.16GHz, 2159 Mhz, 2 Core(s) and 2.00 GB RAM, running Microsoft Windows 7 Ultimate. And for the different mining functions, we use the WEKA tool version 3.9.2 (the development version).

Data Sets

To test the effectiveness of the four aggregation methods, we have used five datasets from OpenML (Open Machine Learning) domain. These datasets have a various number of features and instances. Table 1 summarizes the characteristics of the selected datasets. To attain general results, the datasets are selected in a way that reflects their diversity in the number of features, instances, and type.

Table 1: Characteristic of the Datasets

Dataset Name	Number of Features	Number of Instances
Credit-g	20	1000
Ozone-level-8hr	72	2534
Madelon	500	2600
Micro-mass	1300	571
Bioresponse	1776	3651

Filter Feature Selection Methods

Four classical feature selection methods that are widely used by the researchers in this field were chosen to be applied in this study. Namely; Correlation, InfoGain, GainRatio, and OneR. All of them are available in the WEKA tool.

Aggregation Methods

Although many aggregation methods have been proposed and used, we choose to investigate the efficiency of the well-known, fast, and simple aggregation methods are shown in table (2).

Table (2): Aggregation Methods

Aggregation Method	Meaning
Min	Assigning to each element to be ranked the minimum (best) position that it has achieved among all rankings
Median	Assigning to each element to be ranked the median of all the positions that it has achieved among all rankings
Arithmetic Mean	Assigning to each element to be ranked the mean of all the positions that it has achieved among all rankings
Geometric Mean	Assigning to each element to be ranked the geometric mean of all the positions that it has achieved among all rankings

Classifiers

To evaluate the effectiveness of aggregation methods, we test the performance of the ensemble models using nine machine learning classifiers from different classes (Trees, Rules, and Bayes), shown in table (3). The classes and classifiers are selected randomly for the generality of results.

Table (3): The Classifiers and their Classes

Class Name	Classifiers
Tress	Trees.RandomForest
	Trees.J48
	Trees.RandomTree
Rules	Rules.DecisionTable
	Rules.JRip
	Rules.OneR
Bayes	Bayes.BayesNet
	Bayes.NaiveBayes
	Bayes.NaiveBayesMultinominalText

The Rank Combination Tool (RCT)

The WEKA tool does not contain a function to aggregate ranked lists, so we develop a tool to combine ranked lists. The interface of the developed tool is shown in figure (1).

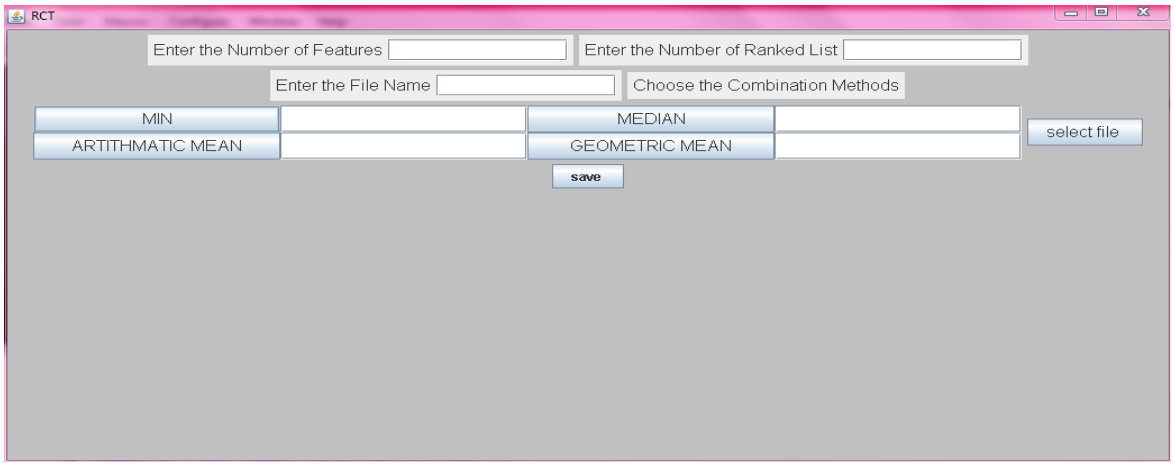


Fig. (1): Rank Combination Tool (RCT) GUI

Number of Top Listed Features

Since base selectors (Filter Methods) and the constructed ensemble models produce rank lists for all features, only some of the top listed features will be selected to be used as feature subsets with classifiers. Table (4) shows the number of top listed features that we use for each data set. Since there is no standard method to select the number of top-ranked features, we base the selection on trial and error, i.e. we take the number of features that show the best accuracy with most of the classifiers. We test the top numbers of features 5, 10, 15, ….40.

Table (4): Number of top listed features used

Dataset Name	No. of top selected features
Credit-g	10
Ozone-level-8hr	15
Madelon	20
Micro-mass	25
Bioresponse	25

EXPERIMENTS AND RESULTS

In this study, we form heterogeneous ensemble models using four filter feature selection methods and four aggregation methods. The performance of every single filter along with the ensemble models is evaluated using the 5 datasets and the 9 classifiers shown above. In practical experiments, 10 cross validation is applied where the available data are randomly divided into 10 disjoint subsets of approximately equal size. One of the subsets is then is used as the test set and the remaining nine sets are used for building the classifier.

Due to the restricted length of the paper, we will only present in details how we examine the performance of the individual filters and the ensemble models using the Credit-g dataset. For other data sets we will show only the final results.

Performance of Individual Filters and Ensemble Models using Credit-g Dataset

The Credit-g Dataset contains 20 attributes and 1000 instances. Table (5) shows the ranked lists generated by the individual filter and the four ensemble models that have been constructed using the four aggregation methods.

Table (5): Ranked Lists Generated by Individual Filters and the Ensemble Models for Credit-g Dataset

Filter	Ranked List
CorrelationAttributeEval	1,2,5,6,15,14,13,3,20,4,8,9,12,7,16,19,17,10,18,11
InfoGainAttributeEval	1,3,2,6,4,5,12,7,15,13,14,9,20,10,17,19,18,8,11,16
GainRatioAttributeEval	1,20,3,2,5,6,13,15,14,4,10,12,7,9,19,17,16,8,11,18
OneRAttributeEval	3,2,20,7,8,4,6,19,9,10,11,18,16,12,15,14,17,13,1,5
Ensemble Model	**Ranked List**
Min	1,3,2,20,5,6,7,4,8,15,14,12,13,19,9,10,11,18,16,17
Median	1,2,3,5,6,20,4,13,15,7,14,9,10,12,8,19,16,17,18,11
Arithmetic Mean	2,3,1,6,20,4,5,7,15,13,14,9,12,8,10,19,16,17,11,18
Geometric Mean	1,2,3,6,20,5,4,7,15,13,14,8,9,12,10,19,11,16,17,18

Tables 6, 7, and 8, show the classification accuracy achieved by the 9 classifiers and the average performance for each of the 3 classes of classifiers Trees, Rules and Bayes.

Table (6): Evaluation of the Base Selectors and the Ensemble Models using Trees Class Classifiers

Filter Technique	Machine Learning Algorithm	Correctly Classified Instances	Incorrectly Classified Instances	Average of Correctly Classified Instances
CorrelationAttributeEval+ ranker				
	Trees.J48	72.7 %	27.3 %	
	Trees.RandomForest	75.3 %	24.7 %	
	Trees.RandomTree	67.2 %	32.8 %	
				71.73%
InfoGainAttributeEval+ ranker				
	Trees.J48	72.4 %	27.6 %	
	Trees.RandomForest	76.3 %	23.7 %	
	Trees.RandomTree	67.9 %	32.1 %	
				72.20%
GainRatioAttributeEval+ ranker				
	Trees.J48	72.7 %	27.3 %	
	Trees.RandomForest	75.3 %	24.7 %	
	Trees.RandomTree	67.2 %	32.8 %	
				71.73%
OneRAttributeEval+ ranker				
	Trees.J48	70.6 %	29.4 %	
	Trees.RandomForest	70.7 %	29.3 %	
	Trees.RandomTree	65.3 %	34.7 %	
				68.87%
Ensemble Model using Min				
	Trees.J48	**72.8 %**	**27.2 %**	
	Trees.RandomForest	76.1 %	23.9 %	
	Trees.RandomTree	68 %	32 %	
				72.30%
Ensemble Model using Median				
	Trees.J48	72.1 %	27.9 %	
	Trees.RandomForest	77.1 %	22.9 %	
	Trees.RandomTree	71.2 %	28.8 %	
				73.47%

Ensemble Model using Arithmetic Mean				
	Trees.J48	72.1 %	27.9 %	
	Trees.RandomForest	77.1 %	22.9 %	
	Trees.RandomTree	71.1 %	28.8 %	
				73.43%
Ensemble Model using Geometric Mean				
	Trees.J48	72.1 %	27.9 %	
	Trees.RandomForest	77.1 %	22.9 %	
	Trees.RandomTree	71.2 %	28.8 %	
				73.37%

Table (7): Evaluation of the Individual Filters and the Ensemble Models using Rules Class Classifiers

Filter Technique	Machine Learning Algorithm	Correctly Classified Instances	Incorrectly Classified Instances	Average of Correctly Classified Instances
CorrelationAttributeEval+ ranker				
	Rules.DecisionTable	71.3 %	28.7 %	
	Rules.JRip	71.9 %	28.1 %	
	Rules.OneR	66.1 %	33.9 %	
				69.77%
InfoGainAttributeEval+ ranker				
	Rules.DecisionTable	70.3 %	29.7 %	
	Rules.JRip	72.4 %	27.6 %	
	Rules.OneR	66.1 %	33.9 %	
				69.60%
GainRatioAttributeEval+ ranker				
	Rules.DecisionTable	71.3 %	28.7 %	
	Rules.JRip	71.9 %	28.1 %	
	Rules.OneR	66.1 %	33.9 %	
				69.77%
OneRAttributeEval+ ranker				
	Rules.DecisionTable	70.4 %	29.6 %	
	Rules.JRip	72.2 %	27.8 %	
	Rules.OneR	71.1 %	28.9 %	
				71.23%
Ensemble Model using Min				
	Rules.DecisionTable	**70.7 %**	**29.3 %**	
	Rules.JRip	72.3 %	27.7 %	

	Rules.OneR	66.1 %	33.9 %	
				69.70%
Ensemble Model using Median				
	Rules.DecisionTable	70.6 %	29.4 %	
	Rules.JRip	71 %	29 %	
	Rules.OneR	66.1 %	33.9 %	
				69.23%
Ensemble Model using Arithmetic Mean				
	Rules.DecisionTable	70.6 %	29.4 %	
	Rules.JRip	71 %	29 %	
	Rules.OneR	66.1 %	33.9 %	
				69.23%
Ensemble Model using Geometric Mean				
	Rules.DecisionTable	70.6 %	29.4 %	
	Rules.JRip	71 %	29 %	
	Rules.OneR	66.1 %	33.9 %	
				69.23%

Table (8): Evaluation of the Individual Filters and the Ensemble Models using Bayes Class Classifiers

Filter Technique	**Machine Learning Algorithm**	**Correctly Classified Instances**	**Incorrectly Classified Instances**	**Average of Correctly Classified Instances**
CorrelationAttributeEval+ ranker				
	Bayes.BayesNet	73.2 %	26.8 %	
	Bayes.NaiveBayes	75.4 %	24.6 %	
	Bayes.NaiveBayesMultinominalText	70 %	30 %	
				72.87%
InfoGainAttributeEval+ ranker				
	Bayes.BayesNet	75 %	25 %	
	Bayes.NaiveBayes	74.6 %	25.4 %	
	Bayes.NaiveBayesMultinominalText	70 %	30 %	
				73.20%
GainRatioAttributeEval+ ranker				
	Bayes.BayesNet	73.2 %	26.8 %	
	Bayes.NaiveBayes	75.4 %	24.6 %	
	Bayes.NaiveBayesMultinominalText	70 %	30 %	
				72.87%
OneRAttributeEval+ ranker				
	Bayes.BayesNet	73.2 %	26.8 %	
	Bayes.NaiveBayes	74.4 %	25.6 %	

	Bayes.NaiveBayesMultinominal Text	70 %	30 %	
				72.53%
Ensemble Model using Min				
	Bayes.BayesNet	75.1 %	24.9 %	
	Bayes.NaiveBayes	75.5 %	24.5 %	
	Bayes.NaiveBayesMultinominal Text	70 %	30 %	
				73.53%
Ensemble Model using Median				
	Bayes.BayesNet	74.6 %	25.4 %	
	Bayes.NaiveBayes	75.4 %	24.6 %	
	Bayes.NaiveBayesMultinominal Text	70 %	30 %	
				73.33%
Ensemble Model using Arithmetic Mean				
	Bayes.BayesNet	74.6 %	25.4 %	
	Bayes.NaiveBayes	75.4 %	24.6 %	
	Bayes.NaiveBayesMultinominal Text	70 %	30 %	
				73.33%
Ensemble Model using Geometric Mean				
	Bayes.BayesNet	74.6 %	25.4 %	
	Bayes.NaiveBayes	75.4 %	24.6 %	
	Bayes.NaiveBayesMultinominal Text	70 %	30 %	
				73.33%

The above tables are summarized in Table (9). Tables 10 -13, show a similar summary for the performance of individual filters and ensemble model for the other four data sets shown in Table (1)**.**

Table 9: Performance of Individual Filters and Ensemble Models using Credit-g Dataset

Machine Learning Algorithms	**Correlation**	**InfoGain**	**GainRatio**	**OneR**	**Min**	**Median**	**Arithmetic mean**	**Geometric mean**
Trees Class	71.73	72.20	71.73	68.87	72.30	**73.47**	73.43	73.37
Rules Class	69.77	69.60	69.77	**71.23**	69.70	69.23	69.23	69.23
Bayes Class	72.87	73.20	72.87	72.52	**73.53**	73.33	73.33	73.33

Table 10: Performance of Individual Filters and Ensemble Models using Ozone-level-8hr Dataset

Machine Learning Algorithms	Correlation	InfoGain	GainRatio	One R	Min	Median	Arithmetic mean	Geometric mean
Trees Class	92.28	92.63	90.62	92.74	92.37	**92.78**	92.54	92.52
Rules Class	93.55	93.55	93.42	93.45	93.38	**93.71**	93.30	93.36
Bayes Class	**77.23**	76.28	77.07	76.59	75.24	75.61	75.90	74.93

`

Table 11: Performance of Individual Filters and Ensemble Models using Madelon Dataset

Machine Learning Algorithms	Correlation	InfoGain	GainRatio	One R	Min	Median	Arithmetic mean	Geometric mean
Trees Class	79.76	80.81	80.45	77.85	**81.91**	80.53	74.27	80.64
Rules Class	68.82	68.56	68.91	68.13	**68.99**	68.45	67.14	68.59
Bayes Class	57.58	57.47	57.40	57.28	57.24	57.41	**57.81**	57.47

Table 12: Performance of Individual Filters and Ensemble Models using Micro-Mass Dataset

Machine Learning Algorithms	Correlation	InfoGain	GainRatio	One R	Min	Median	Arithmetic mean	Geometric mean
Trees Class	**64.74**	61.70	54.82	60.71	60.95	63.28	62.70	63.11
Rules Class	42.91	40.40	38.12	39.52	40.22	**43.08**	41.45	41.39
Bayes Class	45.13	43.37	39.58	42.32	44.89	**45.53**	45.30	45.07

Table 13: Performance of Individual Filters and Ensemble Models using Bioresponse Dataset

Machine Learning Algorithms	Correlation	InfoGain	GainRatio	One R	Min	Median	Arithmetic mean	Geometric mean
Trees Class	75.19	75.46	72.57	**75.80**	74.98	75.61	75.22	75.14
Rules Class	74.79	75.04	73.73	71.71	70.59	75.11	**75.27**	75.21
Bayes Class	63.97	62.91	61.38	63.96	64.04	63.98	**64.47**	63.74

Results

From Tables 9-13, we find that out of 15 best performance results (marked by Bold Face in tables) only 4 are corresponding to individual filters (namely Correlation and OneR Filter) and 11 to ensemble models, thus the probability that ensemble models outperform individual filters is about 73%.

Out of the 11 best performance ensemble models, the aggregation methods efficiency is Median (5/11) and Arithmetic Mean (3/11), Min (3/11),. For the Tree class and the Rule class classifiers, the best aggregation method is Median, and for Bayes class classifiers the best aggregation method is the Arithmetic Mean. As the number of features increased from 500 to1300 and then to 1776, the efficient aggregation method changes from Min to Median to Arithmetic Mean. This may suggest that for a very high number of features the efficient aggregation method is the Arithmetic Mean. The overall results prove that no aggregation method is the best for all cases.

CONCLUSION

Ensemble feature selection is a potential technique to improve the robustness and performance of classification and prediction. When constructing ensemble models one decision to take is how to aggregate ranked lists when using filter methods as base selectors. Since there are many proposed aggregation methods, the most appropriate one is the one that optimizes the efficiency of the model.

In this paper, we investigate the efficiency of four rank aggregation methods. The experimentation environment contains a variety of datasets, filters, and classifiers. The Median aggregation method shows the best efficiency but as the number of features increases the Arithmetic Mean becomes more efficient.

References

[1] Ali Asghar, Shahrjooi Haghighi, Hichem Frigui , Xiang Zhang. (2019). Ensemble Feature Selection for Biomarker Discovery in Mass Spectrometry-based Metabolomics, SAC '19, Limassol, Cyprus.

[2] Martin Binder, Julia Moosbauer, Janek Thomas, and Bernd Bischl. (2020). Multi-Objective Hyperparameter Tuning and Feature Selection using Filter Ensembles. In Genetic and Evolutionary Computation Conference (GECCO '20), Cancún, Mexico

[3] H. Liu and L. Yu. (2005). Toward integrating feature selection algorithms for classification and clustering. IEEE Transactions on Knowledge and Data Engineering. 17, 491–502.

[4] M. Dash and H. Liu. (2003). Consistency-based search in feature selection. Artif. Intell., 151, 155–176.

[5] Tsymbal, A., Pechenizkiy, M., Cunningham, P.(2005). Diversity in search strategies for ensemble feature selection. Inf. fusion **6**(1), 1566–2535.

[6] Seijo-Pardo, B., Porto-Díaz, I., Bolón-Canedo, V., Alonso-Betanzos, A. (2017). Ensemble feature selection: homogeneous and heterogeneous approaches. Knowledge. Based Syst. 118, 124–139. *https://doi.org/10.1016/j.knosys.2016.11.017*

[7] Saeys, Y., Abeel, T., Van der Peer, Y. (2008). Robust feature selection using ensemble feature selection techniques. In: Daelemans, W., et al. (eds.) European Conference on Machine Learning (ECML PKDD). LNAI 5212, 313–325.

[8] Bolon- Candeo, V. And A. Alonso – Betanzoes. (2018). Recent Advances in Ensembles for Feature Selection. Springer

[9] B. Seijo-Pardo, I. Porto-D´iaz, V. Bol´on-Canedo, A. Alonso-Betanzo.(2016). Ensemble Feature Selection: Homogeneous and Heterogeneous Approaches. Knowledge-Based Systems ·

.[10] Wald, R.; Khoshgoftaar, T. M.; Dittman, D. J.; Awada, W.; and Napolitano, A. (2012). An extensive comparison of feature ranking aggregation techniques in bioinformatics. In IEEE 13th International Conference on Information Reuse and Integration, 377–384.

[11] Wald, R.; Khoshgoftaar, T. M.; and Dittman, D. (2012). Mean aggregation versus robust rank aggregation for ensemble gene selection. In 11th International Conference on Machine Learning and Applications, 1, 63–69.

[12] David J. Dittman, Taghi M. Khoshgoftaar, Randall Wald, and Amri Napolitano. (2013). Classification Performance of Rank Aggregation Techniques for Ensemble Gene Selection. Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference