Statistical Performance Effect of Feature Selection Techniques on Eye State Prediction Using EEG

Jean de Dieu Uwisengeyimana, Nusaibah Khalid Al_Salihy and Turgay Ibrikci*

Department of Electrical and Electronics Engineering, Çukurova University, Adana, Turkey

Abstract: Several recent studies have demonstrated that electrical waves recorded by electroencephalogram (EEG) can be used to Predict eye state (Open or Closed) and all the studies in the literatures used 14 electrodes for data recording. To reduce the number of electrodes without affecting the statistical performance of an EEG device, it is not an easy task. Hence, the focus of this paper is on reducing the number of EEG electrodes by means of feature selection techniques without any consequences on the statistical performance measures of the earlier EEG devices. In this study, we compared different attribute evaluators and classifiers. The results of the experiments have shown that ReliefF attribute evaluator was the best to identify the two least important features (P7, P8) with 96.3% accuracy. The overall results show that two data-recording electrodes could be removed from the EEG devices and still perform well for eye state prediction. The accuracy achieved was equal to 96.3% with KStar (K*) classifier which was also the best classifier among the 21 tested classifiers in this study.

Keywords: Classification, Statistical performance, Feature Selection, Machine Learning, EEG.

1. INTRODUCTION

The main issue in machine learning is to find out relationship between an input $X = \{x_1, x_2, \ldots, x_M\}$ and an output *Y*. Sometimes the output *Y* is not determined by the complete set of the input features $\{x_1, x_2, \ldots, x_M\}$, instead, it is decided only by a subset of them $\{x_{(1)}, x_{(2)}, \ldots, x_{(M)}\}$, where *m*<*M*. With sufficient data and time, it is fine to use all the input features, including those irrelevant features, to approximate the underlying function between the input and the output [1]. But, to identify relevant feature subsets, dimensionality reduction methods are used [2].

Those dimensionality reduction methods can be broadly classified into two groups: feature extraction such as principal component analysis (PCA) or linear discriminate analysis (LDA) and feature selection such as Relief [3] or FSDD [4]. Feature extraction method reduces the dimensionality by linear or non-linear projection of Q-dimensional vector on to P-dimensional vector (P<<Q) [5]. However, it changes the original physical features and makes features uninterpretable. On the other hand, feature selection reduces the dimensionality by selecting a subset of original variables. Feature selection methods tend to produce less expensive classifiers. The non-selected variables are no longer needed and they are more easily interpretable [6].

1.1. Feature Selection

Identifying effective features to use for building a classification model for a particular task is a big problem in machine learning which can be approached using dimensionality reduction methods [7]. The method of feature selection which aims to choose a small subset of the relevant features from the original ones according to certain relevance evaluation criterion was preferred in this study. There are many potential benefits of feature selection such as reducing the data measurement and storage requirements, reducing training and utilization times, rejecting the curse of dimensionality to improve prediction performance [8, 9]. Some feature selection methods put more emphasis on one benefit than another, but in most papers the focus is mainly on constructing and selecting subsets of features that are useful to build a good predictor [10, 11]. Actually selecting subsets of variables is usually suboptimal for building a predictor, particularly if the variables are redundant [12]. For this reason, the interest of this paper is on finding and ranking all potentially relevant variables of which the least important ones could be removed.

A general block diagram of feature selection for classification task is shown in Figure **1** [13]. Instead of processing data with the whole features to the learning algorithm directly, feature selection will be performed first to select a subset of features and then process the data with the selected features to the learning algorithm. With the finally selected features, a classifier is induced for the prediction phase.

1.2. Feature Selection Algorithms and Feature Ranking

According to whether the training set is labeled or not, feature selection algorithms can be categorized

*Address correspondence to this author at the Department of Electrical and Electronics Engineering, Çukurova University, Adana, Turkey; Tel: +90(322) 338 68 68; Fax: +90(322) 338 6326; E-mail: ibrikci@cu.edu.tr

E-ISSN: 1929-6029/16

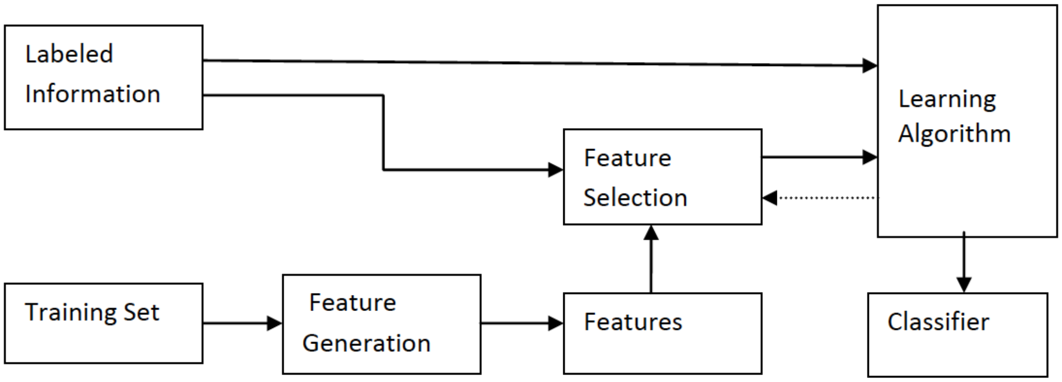

Figure 1: A general block diagram of feature selection for classification.

into supervised, unsupervised and semi-supervised feature selection [14]. A feature selection algorithm can be seen as the combination of a search technique for proposing new feature subsets, along with an evaluation measure which scores the different feature subsets. Many feature selection algorithms include variable ranking as a principal or auxiliary selection mechanism because of its simplicity, scalability, and good empirical success [15]. Following the classification in [16], variable ranking is a filter method: it is a preprocessing step, independent of the choice of the predictor.

2. MATERIALS AND METHODS

2.1. Descriptions of Data Sets

The datasets used in this study are publicly available at “The Data Mining Repository of University of California Irvine (UCI)” [17]. All data is from one continuous EEG measurement with the Emotive EEG Neuroheadset. Emotive headset is a device which gives the value at each instance and is shown in Figure **2**.

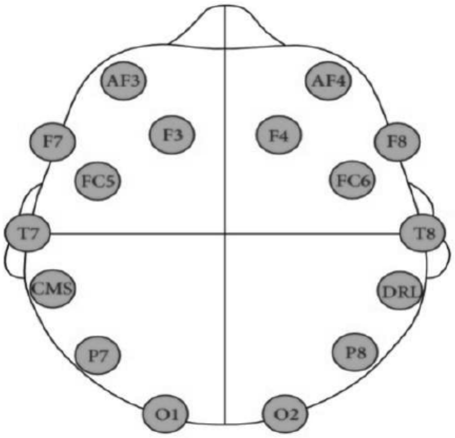

Figure 2: Emotive EEG Neuroheadset electrode position and corresponding behavior groups [18].

The device is composed of 16 electrodes named as F7, F3, F4, FC6, T8, P8, O2, CMS for eye open state and AF3, AF4, FC5, F8, T7, P7, O1, DRL for eye closed state. These electrodes are discs that conduct electrical activity. They capture it from the brain and conduct it through a wire to a machine that amplifies the signal [19]. The duration of the data measurement was 117 seconds where the eye state was detected via a camera during the EEG measurement and added later manually to the file after analyzing the video frames. '*1*' indicates the eye-closed and '*0*' the eye-open state.

The corpus consists of 14977 instances with 15 attributes each (14 attributes representing the values of the electrodes and the eye state). Table **1** shows the value ranges of the 14 electrodes in the corpus [20].

From Table **1**, there is an obvious difference in amplitude of certain electrodes when comparing the range of values for different eye states. On the one hand, for the electrodes F7, F3, O2, P8, T8, FC6, and F4, the maximum values for the eye open state are higher than the maximum values of the eye closed state while the minimum values are nearly the same. On the other hand, for the electrodes AF3, FC5, T7, P7, O1, F8, and AF4, the minimum values for the eye open state are lower than for the eye closed state while the maximum values are about the same.

All what electrodes have in common is that open eye state comes along with a higher value range than the eye closed state while the mean stays nearly the same. Accordingly, also the standard deviation increases.

2.2. Feature Selection and Learning Algorithms in Weka

Weka has been used for feature selection and for classification in this study [21]. The process of Feature

Table 1: Ranges and Means of The electrode Values for The Eye States

Eye State	Eye- closed				Eye- Open			
Electrodes	Min	Mean	Max	Std	Min	Mean	Max	Std
AF3	4198	4305	4445	33.46	1030	4297	4504	54.27
F7	3905	4005	4138	27.54	3924	4013	7804	52.37
F3	4212	4265	4367	20.13	4197	4263	5762	27.66
FC5	4058	4121	4214	21.31	2453	4123	4250	27.58
T7	4309	4341	4435	18.08	2089	4341	4463	29.54
P7	4574	4618	4708	17.44	2768	4620	4756	28.06
O1	4026	4073	4167	24.14	3581	4071	4178	18.60
O2	4567	4616	4695	18.44	4567	4615	7264	34.38
P8	4147	4202	4287	18.55	4152	4200	4586	17.69
T8	4174	4233	4323	19.36	4152	4229	6674	33.48
FC6	4130	4204	4319	23.71	4100	4200	5170	27.08
F4	4225	4281	4368	18.57	4201	4277	7002	36.62
F8	4510	4610	4811	32.79	86	4601	4833	59.88
AF4	4246	4367	4552	34.82	1366	4356	4573	52.28

Selection is separated into two parts: Attribute or Feature Evaluator and Search Method. Attribute evaluator, is the method by which a subset of attributes are assessed, while the Search Method is the structured way in which the search space of possible attribute subsets is navigated based on the subset evaluation. After deciding the best Feature evaluator, we used 21 classifying algorithms to build the classifying model.

3. RESULTS AND DISCUSSIONS

In the earlier study " A First Step towards Eye State Prediction Using EEG " conducted by Oliver Rosler and David Suendermann who used all 14 features they found out that KStar was the best on this dataset with 96.7% of accuracy [20]. In this study, we intended to decrease the number of features by means of Feature Selection techniques. Filter methods which statistically estimate a score or a rank for each feature were preferred over wrapper methods which score subset of features [22]. The Attribute Evaluators and respective ranks given to features are shown in Table **2**, where it is seen that ReliefF Attribute Evaluator was the best to identify the least scored features which can be manually removed from features before using a classifying algorithm.

As shown in Table **3**, different Attribute Evaluators were compared by removing one attribute at a time and noting how it affected the performance of the KStar classifier. Again, among the different Attribute Evaluators compared, the ReliefF attribute evaluator along with the ranker search method were shown up the best and it had accuracy of 96%, 96.3% and 95.5% after removing one least important feature (FC5), two least important features (P8, FC5) and three least important features (P7, P8, FC5) respectively.

From Table **3**, it can be seen that the accuracy increased after removing the second feature (P8), which shows that it is a redundant one and pulls down the accuracy. After removing the two least ranked features, the remaining 12 features are the most effective ones which could represent 12 electrodes of an EEG device instead of 14 electrodes.

ReliefF is certainly the best attribute evaluator in binary classification. It was proposed by Kira and Rendell in 1992 [23]. Its strengths are that it focuses on arriving to optimal solution while most other attribute evaluators are concerned with the quickness of computing. Also, ReliefF is noise-tolerant, robust to feature interactions and is good for binary or continuous data.

In this study, correlation as an attribute evaluator method had least performance because it is dependent on heuristics. It measures the correlation (Pearson's) between a feature and a class. Considering the prediction of a continuous outcome *y*, the Pearson correlation coefficient *R(i)* is defined as:

Table 2: Attribute Evaluators and Respective Ranked Attributes

	ReliefF AttributeEval Ranked Attributes		GainRatio AttributeEval RankedAttributes		Correlation AttributeEval Ranked Attributes		InfoGain AttributeEval Ranked Attributes		OneR AttributeEval Ranked Attributes	
1	0.00054402	F7	0.02920	AF3	0.079994	F7	0.0598	O1	0.626035	O1
2	0.00041021	FC6	0.02454	AF4	0.064294	FC6	0.0572	P7	0.619092	P7
3	0.00035232	T7	0.02430	O1	0.047965	F4	0.0493	AF3	0.605941	AF3
4	0.00031124	T8	0.02380	FC6	0.047218	T8	0.0460	AF4	0.592724	AF4
5	0.00030334	F4	0.02183	P7	0.038902	F3	0.0323	F8	0.583511	F8
6	0.00028527	F3	0.01734	F4	0.025100	O2	0.0257	F4	0.581442	P8
7	0.00014065	O2	0.01387	T7	0.013120	F8	0.0223	P8	0.574900	T8
8	0.00004284	F8	0.01306	T8	0.010458	AF3	0.0216	T8	0.566956	F4
9	0.00002124	AF3	0.01276	F8	0.009576	P8	0.0194	FC6	0.563151	T7
10	0.00001303	O1	0.01255	P8	0.007845	P7	0.0174	T7	0.561482	F3
11	0.00001018	AF4	0.00998	F7	0.007550	AF4	0.0150	O2	0.560080	FC5
12	0.00000944	P7	0.00933	FC5	0.007531	FC5	0.0130	FC5	0.557744	O2
13	0.00000708	P8	0.00787	O2	0.007223	O1	0.0129	F7	0.557076	FC6
14	0.00000579	FC5	0.00733	F3	0.000369	T7	0.0124	F3	0.542924	F7

Table 3: Comparison among the Ranked Attribute Evaluators with KStar Classifier

Attribute Evaluator (With a Ranker Search Method)	No of Removed Attributes	Name of Removed Attributes	TP Rate	FP Rate	MCC	ROC Area	PRC Area
-	0	-	0.968	0.033	0.935	0.995	0.995
Correlation Attribute Evaluator	1	T7	0.960	0.042	0.918	0.993	0.993
	2	O1, T7	0.939	0.062	0.878	0.984	0.984
	3	FC5, O1, T7	0.919	0.084	0.836	0.976	0.976
InfoGain Attribute Evaluator	1	F3	0.963	0.039	0.924	0.994	0.994
	2	F7, F3	0.947	0.055	0.894	0.989	0.989
	3	FC5, F7, F3	0.935	0.068	0.868	0.983	0.983
GainRatio Attribute Evaluator	1	F3	0.963	0.039	0.924	0.994	0.994
	2	O2, F3	0.960	0.042	0.918	0.993	0.993
	3	FC5, O2, F3	0.949	0.053	0.897	0.989	0.990
ReliefF Attribute Evaluator	1	FC5	0.960	0.041	0.919	0.993	0.993
	2	P8, FC5	0.963	0.038	0.925	0.994	0.994
	3	P7, P8, FC5	0.955	0.046	0.909	0.992	0.992
OneR Attribute Evaluator	1	F7	0.955	0.047	0.909	0.991	0.991
	2	FC6, F7	0.941	0.061	0.881	0.986	0.986
	3	O2,FC6, F7	0.930	0.074	0.858	0.981	0.982

$$R(i) = \frac{\sum_{k=1}^{m} (x_{k,i} - \bar{x}_i)(y_k - \bar{y})}{\sqrt{\sum_{k=1}^{m} (x_{k,i} - \bar{x}_i)^2 \sum_{k=1}^{m} (y_k - \bar{y})^2}} \qquad (1)$$

where the bar notation stands for an average over the index *k*. Note that, Correlation criteria such as *R*(*i*) can only detect linear dependencies between variable and target. And, *m* shows the number of points.

After feature selection process, we performed feature classification based on 21 different machine learning algorithms. They were applied and compared to build an eye state classifying model and obtained results for various statistical performance measures such as TP rate (TP), FP rate (FP), Matthews correlation coefficient (MCC), Receiver Operating Characteristic (ROC) area and Precision-Recall curve area (PRC) were calculated and are presented in the Table **4**. It should be noted that Table **4** uses the 11 most effective features by excluding the 3 least important features selected by the ReliefF Attribute Evaluator*.*

Precision (specificity) and recall (sensitivity) are also other basic measures used in evaluating the models which can be calculated from the below formulas:

Recall: The proportion of actual positives which are predicted positive. It is the fractions of relevant instances that are retrieved instances.

$$Recall(Sensitivity) = \frac{TP}{TP + FN} \qquad (2)$$

Precision: The proportion of predicted positives which are actual positive. It is the fraction of retrieved instances that are relevant in recognition with binary classification. It is also known as positive predicted value.

$$Precision(Specificity) = \frac{TP}{TP + FP} \qquad (3)$$

where in the formulas above and in the Table **4**, *TP is* true positive and *FP is* false positive.

F-measure: It is harmonic mean between Precision and Recall and also known as F-score. It considers both the precision and the recall of the test to compute the score. Precision is the number of correct positive results divided by the number of all positive results, and recall is the number of correct positive results divided by the number of positive results that should have been returned. The F-score can be interpreted as a weighted average of the precision and recall [24].

$$F = 2 * \frac{Precision * Recall}{Precision + Recall} \qquad (4)$$

Table 4: The Statistical Performance Averages for the 21 Different Classifiers Trained with 11 Features

	Algorithm	TPRate	FP ate	MCC	ROCArea	PRC Area
1	KStar	0.955	0.046	0.909	0.992	0.992
2	Random Forest	0.929	0.076	0.857	0.982	0.982
3	Random Committee	0.911	0.098	0.822	0.971	0.965
4	Bagging	0.887	0.120	0.772	0.956	0.957
5	Random Sub Space	0.857	0.138	0.744	0.949	0.951
6	LMT	0.867	0.157	0.711	0.938	0.940
7	Classification Via Regression	0.851	0.156	0.698	0.925	0.921
8	J48	0.838	0.165	0.673	0.854	0.815
9	IBK(B1)	0.834	0.170	0.665	0.832	0.779
10	RandomTree	0.834	0.171	0.664	0.832	0.778
11	PART	0.837	0.172	0.669	0.904	0.885
12	REPTree	0.822	0.185	0.639	0.872	0.853
13	Filtered Classifier	0.761	0.257	0.515	0.820	0.811
14	Attributed Selected Classifier	0.712	0.305	0.414	0.767	0.751
15	Decision Table	0.725	0.305	0.442	0.798	0.797
16	Logit Boost	0.679	0.347	0.344	0.723	0.717
17	Bayes Net	0.650	0.376	0.284	0.703	0.695
18	Multi Layer Perceptron	0.556	0.479	0.082	0.565	0.567
19	Simple Logistic	0.582	0.469	0.131	0.602	0.617
20	Multi Class Classifier	0.588	0.454	0.148	0.609	0.627
21	Voted Perceptron	0.552	0.545	0.020	0.504	0.508

Another important evaluation measurement also shown in Table **4** for all classifiers is Receiver Operating Characteristic (ROC). In statistics, ROC curve is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied. The curve measures the classifier's skill in ranking a set of patterns according to the degree to which they belong to the positive class, but without actually assigning patterns to classes. In ROC curve, *recall* and *(1- precision)* are plotted on two axes by using values. Thus, each point on the ROC curve represents a recall/precision pair corresponding to a particular decision threshold [25].

Also, from Table **4**, different types of classifiers such as tree based, lazy based, rule based, meta based, Bayesian based and function based have been applied. It follows that the first 10 classifiers belonged to Lazy, meta and tree groups and had the error rate below to 15%, whereas, standard classifiers in Bayesian, function and rule groups such as bayesNet, MLP, Simple Logistic which have usually high classification performance produced rather poor results on this task (over 30% classification error) see Figure **3**.

Among the all classifiers the best on this eye state dataset was KStar with a classification error of about 4%. KStar is an instance-based classifier under lazy group in Weka. "Instance-based" means that the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function. It differs from other instance-based learners in that it uses an entropy-based distance function [26].

In Figure **3** (below) the classification error rates of the classifiers tested are shown. The error rates in Figure **3** are almost the same for 11 features after feature selection in this study as compared to the results obtained by Oliver Rosler and David Suendermann [20] in their study.

4. CONCLUSION

In this paper, different techniques and algorithms for feature selection were applied on the Eye state datasets available at UCI Machine Learning Repository website. The purpose was to reduce the number of eye state data recording electrodes in EEG devices. Five feature evaluators were compared to find out which was the best on these datasets. After identifying the best feature selection algorithm which was ReliefF Attribute Evaluator, twenty-one classifiers were compared to build an eye state prediction model with

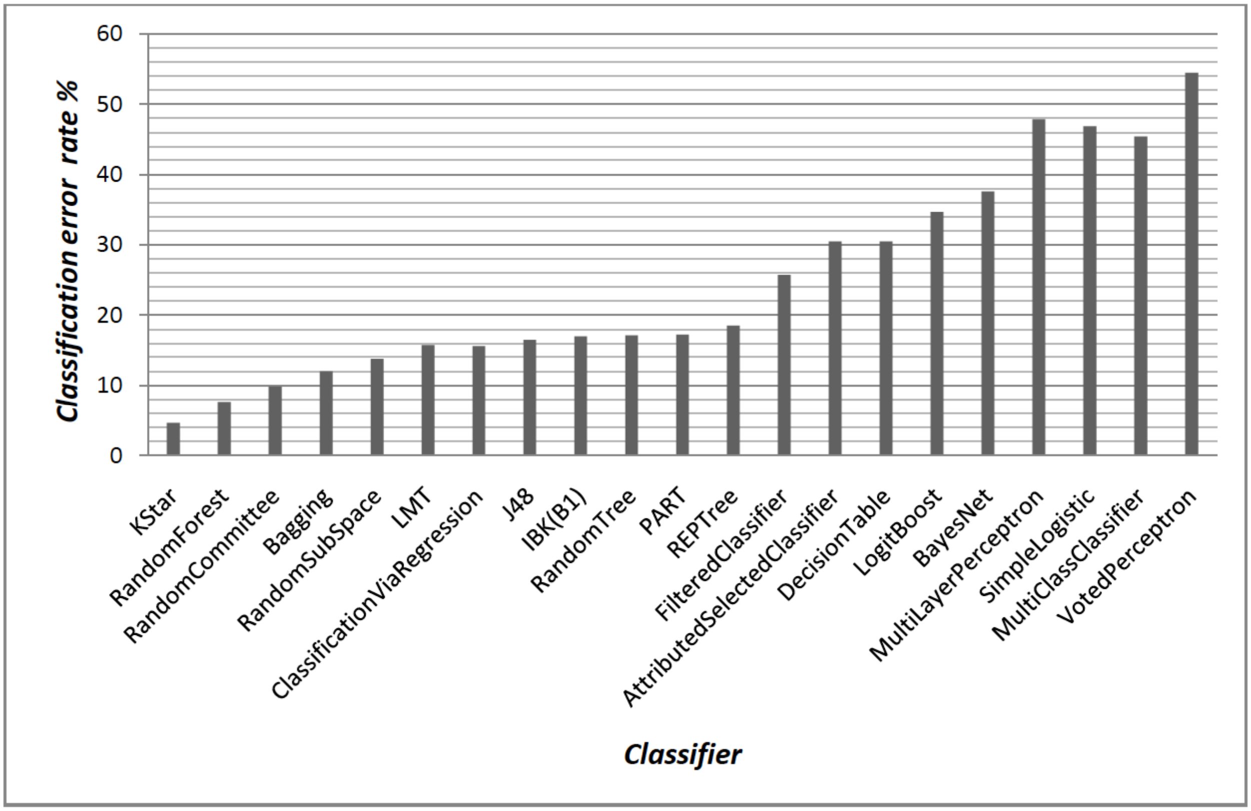

Figure 3: Classification error rate of all classifiers after feature selection process.

reduced features. The results of statistical performance measures for this experiments have shown that two electrodes , identified through the process of variable selection and ranking, could be omitted from the EEG device with an accuracy of 96.3% with KStar(K*) classifier which outperformed other classifiers. Hence, this way is recommended as it reduces production cost of required EEG devices and also speeds up instance-based classification without impact on the performance of the devices.

ACKNOWLEDGMENTS

The authors would like to thank the Scientific and Technological Research Council of Turkey (TÜBİTAK)-BIDEB2235 for their funds which made possible this work. The authors would also like to thank colleagues from University of Rwanda and Çukurova University for comments and suggestions that have resulted in a much improved manuscript.

REFERENCES

[1] Fabio MP, Alexis M, Emiro F, *et al.* Feature selection, learning metrics and dimension reduction in training and classification processes in intrusion detection systems. Journal of Theoretical and Applied Information Technology 2015; 82(2).

[2] Vipin K, Sonajharia M. Feature Selection: A literature Review. Smart Computing Review 2014; 4(3).

[3] Durgabai RP. Feature Selection using ReliefF Algorithm. International Journal of Advanced Research in Computer and Communication Engineering 2014; 3(10).

[4] Liang J, Yang S, Winstanley A. Invariant optimal feature selection: A distance discriminant and feature ranking based solution. In: Pattern Recognition 2008; 41(5): 1429-1439. http://dx.doi.org/10.1016/j.patcog.2007.10.018

[5] Arunasakthi K, Kamatchi PL. A Review on Linear and Non-Linear Dimensionality Reduction Techniques. Machine Learning and Applications: An International Journal (MLAIJ) 2014; 1(1).

[6] Andreas J, Wilfried NG. On the Relationship between Feature Selection and Classification Accurac. JMLR: Workshop and Conference Proceedings 2008; 4: 90-105.

[7] Mark AH. Correlation-based Feature Selection for Machine Learning. Thesis of Doctor of Philosophy at The University of Waikato, April 1999.

[8] Albert S, Francisco JR, Andreu C, *et al.* Interval-valued Feature Selection. CETpD, Neàpolis Building. Rambla de l'Exposició, pp. 59-69.

[9] Gianluca B. On the use of feature selection to deal with the curse of dimensionality in microarray data Available at http://www.ulb.ac.be/di/map/gbonte/ftp/gand.pdf, Machine Learning Group Université Libre de Bruxelles.

[10] Wasif A, Richard T. Towards benchmarking feature subset selection methods for software fault prediction. Computational Intelligence and Quantitative Software Engineering 2016; 617: 33-58. http://dx.doi.org/10.1007/978-3-319-25964-2_3

[11] Ramaswami M, Bhaskaran R. A Study on Feature Selection Techniques in Educational Data Mining. Journal of Computing 2009; 1(1).

[12] Isabelle G, Andre E. An Introduction to Variable and Feature Selection. Journal of Machine Learning Research 2003; 3: 1157-1182.

[13] Jiliang T, Salem A, Huan L. Feature selection for classification: A review. In: Data Classification: Algorithms and Applications. CRC Press 2014; p. 37.

[14] Zhao Z, Liu H. Semi-supervised feature selection via spectral analysis. Proceedings of SIAM International Conference on Data Mining 2007; pp. 641-646. http://dx.doi.org/10.1137/1.9781611972771.75

[15] Ronaldo CP. Combining feature ranking algorithms through rank aggregation. The 2012 International Joint Conference on Neural Networks (IJCNN) 2012; pp.1-8.

[16] Kohavi, John G. Wrappers for feature selection. Artificial Intelligence 1997; 97(1-2): 273-324. http://dx.doi.org/10.1016/S0004-3702(97)00043-X

[17] Lichman M. UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science 2013. Available at http://archive.ics.uci.edu/ml.

[18] Matthieu D. Performance of the EmotivEpoc headset for P300-based applications. Biomed Eng Online 2013; 12: 56. http://dx.doi.org/10.1186/1475-925X-12-56

[19] Sahu M, Nagwani NK, Shrish V, Saransh S. Performance Evaluation of Different Classifier for Eye State Prediction Using EEG Signal. International Journal of Knowledge Engineering 2015; 1(2): 141-145. http://dx.doi.org/10.7763/IJKE.2015.V1.24

[20] Oliver R, David S. A First Step towards Eye State Prediction Using EEG. Baden-Wuerttemberg Cooperative State University (DHBW), Germany 2013.

[21] Ian HW, Eibe F, Mark AH. Data Mining-Practical Machine Learning Tools and Techniques. The Morgan Kaufmann series in data management systems, third Edition 2011.

[22] Vijayasankari S, Ramar K. Enhancing Classifier Performance Via Hybrid Feature Selection and Numeric Class Handling-A Comparative Study. International Journal of Computer Applications 2012; 41(17): 0975-08887.

[23] Kira K, Rendell LA. A practical approach to feature selection. Machine Learning 1992; 249-256. http://dx.doi.org/10.1016/b978-1-55860-247-2.50037-1

[24] Matthias K. Performance Measures in Binary Classification. International Journal of Statistics in Medical Research 2012; 79-81.

[25] Turgay I, Esra MK, Uwisengeyimana JD. Meta Learning on small biomedical datasets. Information Science and Applications (ICISA 2016), Lecture Notes in Electrical Engineering 2016; 376: 933-939. http://dx.doi.org/10.1007/978-981-10-0557-2_89

[26] John GC, Leonard ET. "K*: An Instance-based Learner Using an Entropic Distance Measure. 12th International Conference on Machine Learning 1995; 108-114.

Received on 10-06-2016 Accepted on 29-07-2016 Published on 16-08-2016

http://dx.doi.org/10.6000/1929-6029.2016.05.03.9