CERTIFICATE

This is to certify that the following paper was presented online during the
12th International Conference on Soft Computing and Pattern Recognition (SOCPAR 2020)
held on the world wide web during December 15 - 18 , 2020.

Paper ID: 32

Paper Title: Reducing the Error Mapping of the Students' Performance using Feature Selection

List of Authors: Yuni Yamasari, Naim Rochmawati, Anita Qoiriah, Dwi F. Suyatno and Tohari Ahmad

Springer

Prof. Dr. Ajith Abraham
SOCPAR 2020 - General Chair

Issued On: December 17, 2020

Reducing the Error Mapping of the Students' Performance Using Feature Selection

Yuni Yamasari[1](✉), Naim Rochmawati[1], Anita Qoiriah[1], Dwi F. Suyatno[1], and Tohari Ahmad[2]

[1] Department of Informatics, Universitas Negeri Surabaya, Surabaya, Indonesia
{yuniyamasari,naimrochmawati,anitaqoiriah,
dwifatrianto}@unesa.ac.id
[2] Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
tohari@if.its.ac.id

Abstract. In an educational environment, classifying the cognitive aspect of students is critical. It is because an accurate classification is needed by a lecturer to take the right decision for enhancing a better educational environment. To the best of our knowledge, there is no previous research that focuses on this classification process. In this paper, we propose discretization and feature selection methods before the classification. For this purpose, we adopt the equal frequency for the discretization whose result is evaluated by using logistic regression with two regularizations: lasso and ridge. The experimental result shows that four-intervals on the ridge achieve the highest accuracy. It is to be the base to determine the level of the student's performance: excellent, good, fair, and poor. Next, we remove unnecessary features, by using the Gain Ratio and Gini Index. Also, we build classifiers to evaluate our proposed methods by using k-Nearest Neighbors (k-NN), Neural Network (NN), and CN2 Rule Induction. The experimental result indicates that both discretization and feature selection can enhance the performance of the classification process. Concerning the accuracy level, there is an increase of about 35%, 2.14%, and 3.8% on average of k-NN, NN, and CN2 Rule Induction respectively, from those with original features.

Keywords: Data mining · Features selection · Classification · Performance · Student

1 Introduction

In an educational environment, learning and evaluation systems based on the web are a consequence of the fast growth of Information and Communication Technology (ICT). In this digital era, a web-based learning system is developed to support the learning process. So, students can learn about the subjects they take anytime and anywhere. Also, a web-based evaluation system is built to evaluate the understanding and capability of students in a certain subject. All systems are developed to simplify the learning and evaluation processes, regarding time, cost, and easiness [1].

A. Abraham et al. (Eds.): SoCPaR 2020, AISC 1383, pp. 176–185, 2021.
https://doi.org/10.1007/978-3-030-73689-7_18

The stored data in those systems consist of many features that represent the characteristics of students. They are useful for users to do further processing which is done by data mining. There are many tasks in data mining, including classification, clustering and association analysis, etc. [2]. In the data mining environment, tasks applied to educational data are called educational data mining (EDM). Here, EDM uses statistical, data mining methods, and machine learning [3] for obtaining the required information. Therefore, it can be inferred that EDM focuses more on methodologies and techniques [4].

There is much research relating to the student data processing using machine learning methods as a subset of Artificial Intelligence [5, 6]. In the beginning, the previous research exploits student data to classify sex, age, family size to some groups by applying machine learning methods for the classifiers like Nearest Neighbors, Support Vector Machines, and Naive Bayes algorithms [7]. In other research, classifying student cognitive is done by using Bayesian Network, Naïve Bayes, and J48 [8]. Decision Tree as a classification method is implemented to assess students' skills relating to the designing and conducting experiments within a complex systems micro world [9]. Another research also uses a Decision Tree to mine student data [10]. The goal of this research is to explore Students' Self-Regulation in Learning Contexts. In further research, a binary classifier is built to predict performance on student activity patterns in Virtual Learning Environment (VLE) [11].

All the previous research depicts that classification has been an important factor in EDM. As far as we know, there has been no research focusing on improving the accuracy of classification tasks, whereas, accurate classification of student data gives advantages for the education field. For example, the classification results make it easier for the users to support the best decision because the main goal of EDM is to extract educational data to produce information for supporting the decision making in the educational system [4]. For example: identifying students based on their learning achievement, modeling their achievement, and predicting their learning results. Moreover, this accurate information can be used to design a better academic environment.

Because of those factors, to achieve higher accuracy, much research has worked on the enhancement of the classification performance. Ramaswami et al. study techniques to select certain features to find the best predictive performance of the student achievement classification [12]. Here, they use Correlation, Chi-Square, Gain-Ratio, Information-Gain, Relief, and Symmetrical Uncertainty which are executed with some classifiers, such as Voted Perceptron, Naïve Bayes, one R, and PART. In this case, the best results are generated by information gain and correlation with ROC value = 0,729 and F1-Measure = 0.592. A comparison of feature selection is also made in [13] which explores the student performance classification. Various methods are applied in this research: Genetic Algorithm, Support Vector Machine, Information Gain, and Minimum Redundancy and Maximum Relevance (MRMR). Then, four classifiers are operated to evaluate these techniques: D-tree, Naïve Bayes, k-NN, and ANN. The result shows that MRMR and k-NN are the best methods whose accuracy is 91.12. Regha et al. study about selecting features in educational data mining by using a new technique called Artificial Fish Swarm-Cuckoo Search Optimization. It is done to remove redundancy and irrelevant feature, so the classification performance is more effective [14]. Here, all

research does not explain the effect of the accuracy level on student modeling, especially on the classification.

By considering the scope of previous research, here we intend to work on those problems by proposing methods to optimize the classification process and to evaluate its effects on the classification of student's domain cognitive. Furthermore, different from existing research, we investigate not only increasing the accuracy by implementing the Gini Index and Gain Ratio but also evaluate their effect on the performance of the classification.

2 Method

In this section, we explain our proposed method whose overall process is shown in Fig. 1. It comprises three stages: pre-processing, classification, and post-processing stages. The details of those stages are as follows.

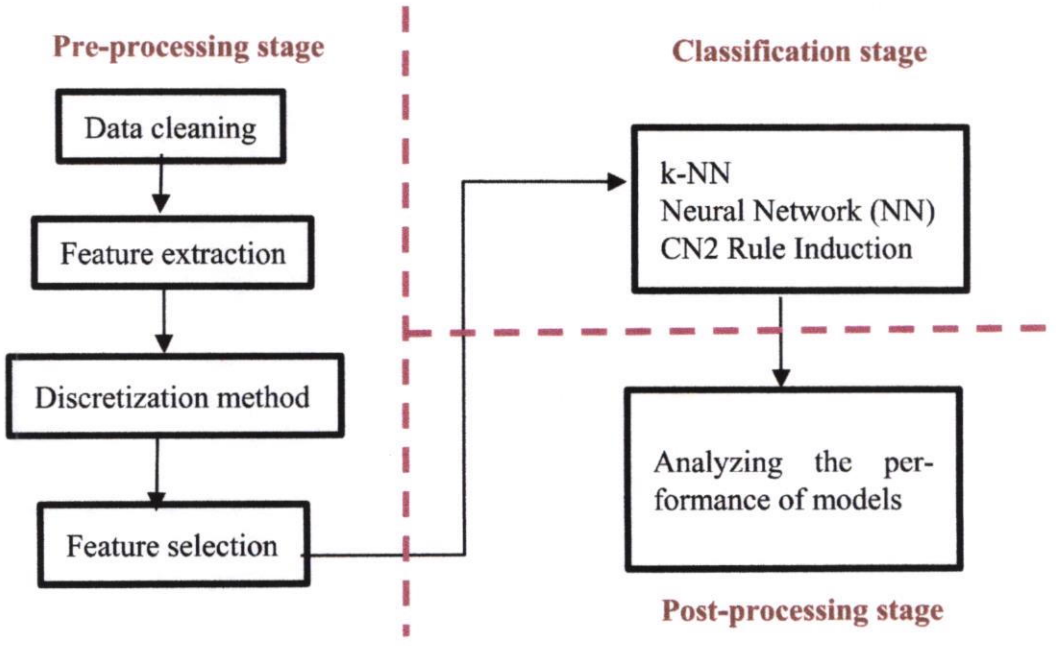

Fig. 1. The proposed method

In the pre-processing stage, stage comprises many steps, namely: data cleaning, category-based features extraction, exploring of discretization method, and feature selection. The data cleaning step is done to avoid redundancy, null variable or feature, noise, etc. for our student data as suggested by [15]. Then, Feature extraction is extracted based on the category, as in our previous research [16] to improve the performance. Discretization Method is explored, namely: equal frequency. We do discretization of student data using three, four, and five-intervals. Then, we build a model by using logistic regression to evaluate this discretization method. Here, there are two regularizations implemented. They are lasso and ridge. In this step, firstly we compute the score of all features, which is then used for evaluating their impact on the dependent (class) variable. For this purpose,

we propose to make use of both the Gini Index and Gain Ratio. In the case of the Gini index, two randomly chosen instances may have different classes. This condition can be depicted in (2), where denote the fraction of records belonging to class i at node t.

$$Gini(t) = 1 - \sum_{i=0}^{c=i} [p(i|t)]^2 \tag{1}$$

$$Entropy(t) = -\sum_{i=0}^{c=i} p(i|t)\log_2 p(i|t) \tag{2}$$

Next, Gain Ratio is defined as the information gain divided by the entropy of the feature's value, and it is introduced by [17]. This research uses the Gain Ratio which is described in (3), (4). In those formulas, and represent the training set, the number of partitions, and the attribute of the data set, respectively.

$$Gain\ Ratio\ (B) = \frac{Gain\ (B)}{Split\ Info\ (B)} \tag{3}$$

$$Split\ Info_B(D) = -\sum_{j=1}^{x} \frac{|D_j|}{|D|} \log_2 \left(\frac{|D_j|}{D} \right) \tag{4}$$

Once the scores of all features have been computed, they are ranked. By considering that some have represented all features, we take of them, with is the number of available features. Therefore, we have 4 out of 10 features.

In the next stage, we construct a classification model by using machine learning: k-NN, Neural Network (NN), and CN2 Rule Induction [18]. Furthermore, we apply Euclidean for the distance parameter, and the number of neighbors is set to 5. Additionally, we design to use a supervised Neural Network with sklearn's Multi-Layers Perceptron (MLP) algorithm which utilizes backpropagation for the training [19, 20]. According to [21, 22], MLP has been able to optimize the log-loss function by using stochastic gradient descent. Lastly, we propose a classification model by using CN2, which is introduced in [23, 24].

Lastly, in the post-Processing Stage, we do analyze the experimental result for investigating whether the previously selected features can produce better performance than that of original features (all features). The performance is measured by utilizing classifier metrics: accuracy, precision, recall, and F1.

3 Result and Discussion

In this section, we analyze the experimental results which are obtained from the proposed method. After data cleaning and feature extraction, student data is generated as depicted in Table 1.

Then, we focus to observe the number of classes. Especially on the classification of student's performance, the number of the class target is very important. Because it indicates the mapping of student's abilities, so we propose the discretization method on the labeling process. Here, we only depict the experimental result on the training set = 60% because almost all intervals achieve the highest accuracy level on this training set.

This is represented in Fig. 2. Additionally, the four-intervals have achieved the Fig. 2. In detail, respectively, the highest accuracy level on the lasso and ridge is reached by three-intervals about 85.9% and by four-intervals about 100%. For that, we decide on four classes on the labeling process for student's performance, namely: poor, fair, good, and excellent.

Table 1. Data type and category-based extraction features

No.	Feature	Data type	Description
1	MID	Numeric	Number of the answered main items
2	MID%_True	Numeric	The proportion of the correct answered to all answered main items
3	Time_MID	Numeric	Time is taken for answering the main item
4	Score_MID	Numeric	A score of the student from the main items
5	GID	Numeric	Number of the answered scaffolding/guide
6	GID%_True	Numeric	The proportion of the correct answered to guide items
7	Time_GID	Numeric	Time is taken for answering the scaffolding/guide items
8	Hint	Numeric	Number of the used hints
9	Score_GID	Numeric	A score of the student from the guide items
10	TotalScore	Numeric	A total score of MID and GID

In the next process, we calculate the score of each feature by implementing the Gini Index and Gain Ratio, as in (2)–(4). In the result using Gini Index, the highest four features are Score_GID, GID, Time_GID, and MID%_True whose value is respectively 0.489, 0.448, 0.41, 0.407. Meanwhile, with Gain Ratio, the highest four features are Hint_GID, Score_GID, GID, and MID%_True having values 0.741, 0.689, 0.613, and 0.613, respectively. For the next process, we ignore the rest features, by assuming that those are irrelevant.

In the process of classification, as described in the previous section, we experiment by using cross-validation. In this research, we apply to 2, 3, 5, 10, and 20 folds. These results are then evaluated by using accuracy, precision, recall, and F1. To investigate the effect of this proposed method, we also implement it to all features (without selecting the feature). We compute the average of all metrics as shown in Table 2. The feature selection can increase significantly the accuracy level average on k-NN by about 24%–26.1%. In contrast, feature selection enhances slightly the accuracy level average on the NN, namely: about 0.4%–2.4%. In the remaining, the feature selection increases moderately the performance on the CN2 Rule Induction method by about 3.9%.

From those three classifiers which have been applied to different sets of features, we only visualize the lowest and the highest of performance as illustrated in Fig. 3(a) and (c). Overall, this figure describes that students' performance is classified into four classes: excellent, good, fair, and poor. It is found that many students are incorrectly classified. This misclassification happens mostly in k-NN with Original_features depicted

in Fig. 3(a). Contrary, when this NN classifier is applied to Gini_Features, there is a significant improvement. That is, the number of misclassified students can be reduced depicted in Fig. 3(c).

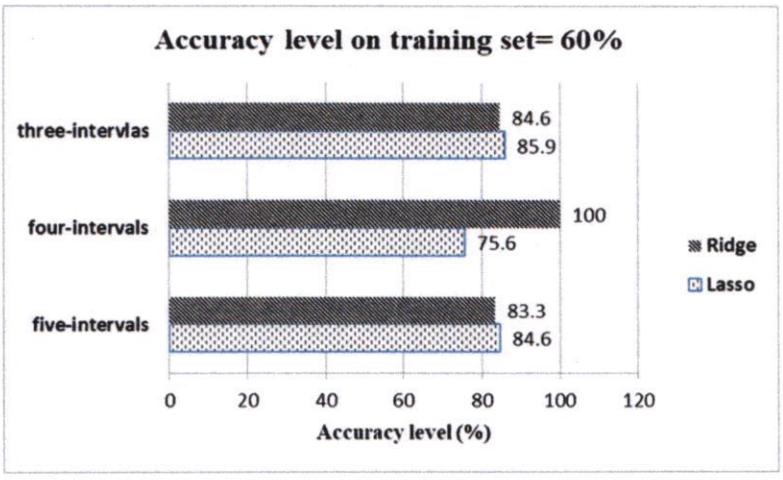

Fig. 2. Comparison of accuracy level on the combination of the discretization method and the logistic regression for all intervals on size of training set = 60%

Table 2. The comparison of performance average on all methods

Method	Accuracy	Precision	F1	Recall
k-NN_original	56.3	0.76	0.73	0.79
k-NN_Gain	92.4	0.97	1	0.94
k-NN_Gini	90.3	0.99	1	0.98
NN_original	92.2	1	1	1
NN_Gain	92.6	0.97	1	0.95
NN_Gini	94.6	1	1	1
CN2 Rule Induction_original	87.9	0.94	0.94	0.93
CN2 Rule Induction_Gain	91.8	0.98	1	0.96
CN2 Rule Induction_Gain	91.8	0.98	1	0.96

For visualizing that classifier performance, we use the ROC (Receiver Operating Characteristics) whose evaluation is performed by calculating its AUC (Area Under Curve). Indirectly, this also expresses the value of the respective confusion matrix. This illustration of the narrowest of AUC occurs on the combination of original features and all classifiers provided in Fig. 3(b). In contrast, the widest of AUC is achieved by the Gini_features on all classifiers illustrated in Fig. 3(d). In detail, a combination of Gini_Features with every classifier is larger than Gain_Features and Original_Features, where CN2 Rule Induction, k-NN, and NN have 0.996, 1, and 1, respectively. On contrary, the smallest AUC is generated by Original_Features. That is, the AUC of k-NN,

NN and CN2 are 0.946, 1, and 0.986. Moreover, we can infer that the use of feature selection (Gain_Features and Gini_Features) can optimally improve the classification performance, especially on k-NN. It is indicated by an increase of AUC by about 0.047–0.054.

For analyzing the error mapping, we observe the best performance of classifiers for analyzing its impact on the students' mapping based on the performance. On k-NN, Gain_Features achieves the best performance on the fold of 10. We compare the number of misclassified students obtaining from Original_Features and Gain_Features. It is shown that Gain_Features reduces the mapping error by 37, from 43 to 6 students.

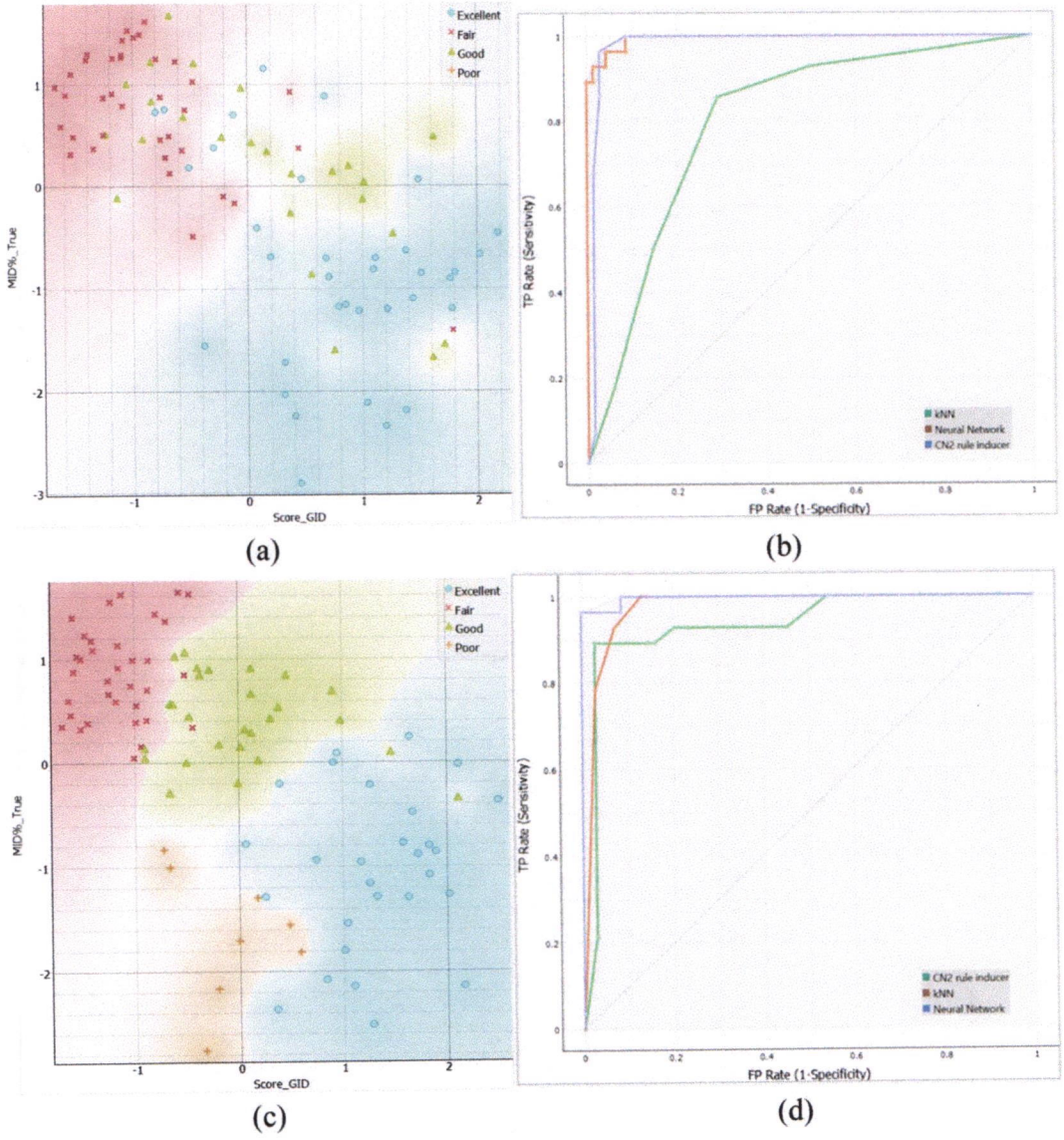

Fig. 3. The visualization of the students' performance mapping on (a) k-NN and Original_features (b) ROC of Original_features on all classifiers (c) NN and Gini features (d) ROC of Gini_features on all classifiers

Gini_Features obtain the best performance on folds of 3 and 5. Gini_Features decreases the error by 34, from 43 to 9, while Gain_Features is by 37, from 43 to 6.

For NN Method, we measure the performance by using NN. It is found that both Gain_Features and Gini_Features have the best performance on the fold of 3. Gain_Features and Gini_Features can reduce the number of misclassified students to 6 and 3, respectively, from 7 students who are generated by Original_Features. It is shown that different from that with k-NN, the combination of NN and Gain_Features holds better results. Lastly, on the CN2 Rule Induction Method, the performance evaluation by using CN2 Rule Induction is done with the fold of 3 since this is better than others. It can be seen that by using CN2 Rule Induction, the number of misclassified students is also lower for both Gain_Features and Gini_Feature than that for Original_Features.

Table 3. Average of accuracy numbers and its corresponding number of students for all sets of features and classification methods

Classification method	Features	Average of accuracy (%)	Number of misclassified students
k-NN	Original_Features	56.3	43
	Gain_Features	92.38	6
	Gini_Features	90.3	9
NN	Original_Features	92.2	7
	Gain_Features	92.58	6
	Gini_Features	94.6	3
CN2 rule induction	Original_Features	87.98	15
	Gain_Features	91.78	7
	Gini_Features	91.78	7

Overall, Gain_Features and Gini_Features can reduce the student mapping errors of all classifiers as provided in Table 3. We can infer from Table 3 that features selection can reduce the mapping error on the students 'performance classification. Overall, the best performance is achieved by NN which is applied to Gini_Features. This combination reaches 94.6% accuracy and produces 3 misclassified students. The second-lowest misclassification, which is 6, is obtained by NN with Gain_Features whose accuracy is 92.58%. On the other hand, k-NN that is applied to Gain_Features also generates 6 misclassified students with slightly lower accuracy, which is 92.38%. It is also shown that CN2 Rule Induction generates lower accuracy for both Gain_Features and Gini_Features. Also, the use of Original_Features leads to the lowest performance, which is represented by lower accuracy and higher misclassification than both Gain_Features and Gini_Features.

4 Conclusion

Our research employs two methods of feature selection: Gain Ratio and Gini Index to reduce the unsuitable mapping on students' performance domain. Those methods have been proven to increase the classifier's performance regarding the Area Under Curve (AUC), they can extend AUC very well. Moreover, feature selection can improve the accuracy level significantly.

References

1. Wanarti, P., Ismayanti, E., Peni, H., Yamasari, Y.: The enhancement of teaching-learning process effectiveness through the development of instructional media based on e-learning of Surabaya's vocational student. In: Proceedings of the 6th International Conference on Educational, Management, Administration and Leadership, pp. 342–346 (2016). https://doi.org/10.2991/icemal-16.2016.71
2. Tan, P.-N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Pearson Addison Wesley, Boston (2005)
3. Dillenbourg, P.: The evolution of research on digital education. Int. J. Artif. Intell. Educ. **26**(2), 544–560 (2016). https://doi.org/10.1007/s40593-016-0106-z
4. Liñán, L.C., Pérez, Á.A.J.: Educational data mining and learning analytics: differences, similarities, and time evolution. RUSC. Univ. Knowl. Soc. J. **12**(3), 98 (2015). https://doi.org/10.7238/rusc.v12i3.2515
5. Koza, J.R., Bennett, F.H., Andre, D., Keane, M.A.: Automated design of both the topology and sizing of analog electrical circuits using genetic programming. In: Gero, J.S., Sudweeks, F. (eds.) Artificial Intelligence in Design 1996, pp. 151–170. Springer, Dordrecht (1996). https://doi.org/10.1007/978-94-009-0279-4_9
6. Samuel, A.L.: Some studies in machine learning using the game of checkers. IBM J. Res. Dev. **3**(3), 210–229 (1959). https://doi.org/10.1147/rd.33.0210
7. Troussas, C., Virvou, M., Mesaretzidis, S.: Comparative analysis of algorithms for student characteristics classification using a methodological framework (2015)
8. Sukajaya, N., Purnama, K.E., Purnomo, M.H.: Intelligent classification of learner's cognitive domain using Bayes Net, Naïve Bayes, and J48 utilizing bloom's taxonomy-based serious game. Int. J. Emerg. Technol. Learn. **10**(2), 46–52 (2015). https://doi.org/10.3991/ijet.v10i1.4451
9. Gobert, J.D., Kim, Y.J., Sao, M.A., Pedro, M.K., Betts, C.G.: Using educational data mining to assess students' skills at designing and conducting experiments within a complex systems microworld. Think. Skills Creativity **18**, 81–90 (2015). https://doi.org/10.1016/j.tsc.2015.04.008
10. Ko, C.-Y., Leu, F.-Y.: Applying data mining to explore students' self-regulation in learning contexts. In: 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), March 2016, pp. 74–78 (2016). https://doi.org/10.1109/AINA.2016.123.
11. Casey, K., Azcona, D.: Utilizing student activity patterns to predict performance. Int. J. Educ. Technol. High. Educ. **14**(1), 4 (2017). https://doi.org/10.1186/s41239-017-0044-3
12. Ramaswami, M., Bhaskaran, R.: A study on feature selection techniques in educational data mining. J. Comput. **1**(1), 2151–9617 (2009). Accessed 16 Aug 2017. https://pdfs.semanticscholar.org/d11c/46515632f3e462d1a952e67fd4657a5f009e.pdf

13. Punlumjeak, W., Rachburee, N.: A comparative study of feature selection techniques for classify student performance. In: 2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE), October 2015, pp. 425–429 (2015). https://doi.org/10.1109/ICITEED.2015.7408984
14. Sasi Regha, R., Uma Rani, R.: Optimization feature selection for classifying student in educational data mining. Int. J. Innov. Eng. Technol. **490**(4) (2016). https://ijiet.com/wp-content/uploads/2017/01/65.pdf. Accessed 16 Aug 2017
15. Han, J., Kamber, M., Pei, J.: Data Mining Concepts and Techniques. Elsevier, USA (2012)
16. Yamasari, Y., Nugroho, S.M.S., Sukajaya, I.N., Purnomo, M.H.: Features extraction to improve performance of clustering process on student achievement. In: 2016 International Computer Science and Engineering Conference (ICSEC), December 2016, pp. 1–5 (2016). https://doi.org/10.1109/ICSEC.2016.7859946
17. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986). https://doi.org/10.1023/A:1022643204877
18. Altman, N.S.: An introduction to Kernel and nearest-neighbor nonparametric regression. Am. Stat. **46**(3), 175–185 (1992). https://doi.org/10.1080/00031305.1992.10475879
19. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323** (1986). https://www.iro.umontreal.ca/~pift6266/A06/refs/backprop_old.pdf
20. LeCun, Y., Bottou, L., Orr, G.B., Muller, K.-R.: Efficient BackProp. Springer, New York (1998)
21. Xu, W.: Towards optimal one pass large scale learning with averaged stochastic gradient descent, July 2011. https://arxiv.org/abs/1107.2490. Accessed 4 April 2018
22. Kingma, D.P., Lei Ba, J.: ADAM: a method for stochastic optimization (2015)
23. Clark, P., Niblett, T.: The CN2 induction algorithm. Mach. Learn. **3**(4), 261–283 (1989). https://doi.org/10.1023/A:1022641700528
24. Clark, P., Boswell, R.: 'Rule induction with CN2: some recent improvements. In: Clark, P., Boswell, R. (eds.) Machine Learning - Proceedings of the 5th European Conference, EWSL-91, pp. 151–163. Springer, Heidelberg (1991)https://www.cs.utexas.edu/users/pclark/papers/newcn.pdf