Feature selection on educational data using Boruta algorithm

Neeyati Anand*, Riya Sehgal,
Sanchit Anand and Ajay Kaushik

Maharaja Agrasen Institute of Technology,
Rohini, Delhi, India
Email: neeyatianand97@gmail.com
Email: riyasehgal0307@gmail.com
Email: sanchitanand97@gmail.com
Email: ajayk08@gmail.com
*Corresponding author

Abstract: Data mining in education deals with formulating strategies for students with the aim to increase the parameters affecting the learning and employability. It also helps the educational institutes in maintaining their reputation as it is directly linked to the student's grades. We need to identify the parameters involved in learning and the relationship among those parameters. In EDM, feature selection (FS) is one of the most important and needed method in EDM, as it removes the features which have no direct link with the student's performance. For example, the date of birth of a student does not impact his/her performance. In this paper, an attempt has been made to improve the performance of the classifiers for undergraduate students at Maharaja Agrasen Institute of Technology. We have applied several techniques of data mining to make some rules that increase the learning and employability of students. The results of our study have shown a significant increase in accuracy, recall, precision and F-measure for naïve Bayes and decision tree classifiers.

Keywords: classifiers; educational data mining; EDM; feature selection; performance.

Reference to this paper should be made as follows: Anand, N., Sehgal, R., Anand, S. and Kaushik, A. (2021) 'Feature selection on educational data using Boruta algorithm', *Int. J. Computational Intelligence Studies*, Vol. 10, No. 1, pp.27–35.

Biographical notes: Neeyati Anand is a final year student. She is pursuing her Bachelor of Technology in Information Technology and is interested in the field of machine learning.

Riya Sehgal is a final year student. She is pursuing her Bachelor of Technology in Information Technology and is interested in the field of machine learning.

Sanchit Anand is a final year student. He is pursuing his Bachelor of Technology in Information Technology and is interested in the field of machine learning.

Ajay Kaushik is an Assistant Professor at the Maharaja Agrasen Institute of Technology, GGSIPU, Delhi. He obtained his Master in Information Technology from USIT, GGSIPU Delhi. He has a teaching experience of 13 years. His research interests include data mining and databases.

1 Introduction

Educational data mining (EDM) helps in finding useful data from the educational datasets. It is critical to understand the student's data so that we can help them and improve their academic performance (Zaffar et al., 2018; Velmurugan and Anuradha, 2016). We can predict their performance by the means of machine learning algorithms and help them by adopting various strategies. Also, there has been an enormous increase in the amount of data available and is still steadily rising. Due to the availability of large datasets we need to clean the messy data and remove irrelevant and redundant information. It can be done with the help of feature selection (FS). FS is a vital step in cleaning the dataset. It enhances the performance of the machine learning algorithm. The raw data contains incoherent information due to which the machine learning algorithm is not able to decipher its meaning (Ramaswami and Bhaskaran, 2001). In this process, only those features are chosen which contribute most to the prediction variable or output in which we are interested. Having irrelevant data decreases the efficiency and the speed of the models. Also, simple machine learning algorithms can be used with simpler data which in turn reduces the overall cost of training the model. The three FS methods are mentioned below:

- *Filter methods:* In this method, the features are not selected based on their importance but are evaluated using various tests that discover correlations with the dependent variable. This method is not dependent on the machine learning algorithm that is being used. Hence, it takes less computational time. Various metrics used are the chi-squared test, correlation metric examples: correlation metrics (pearson, spearman, distance), chi-squared test, Anova, Fisher's Score, etc. (Saurav, 2016).
- *Wrapper methods:* In this method, a subset of features is taken. We get the results by training the model and according to their importance, the features are added or removed from the subset. With the help of this method, the highly unimportant features are removed from the dataset which in turn increases the accuracy of the model. As we are training the model to get the results, it is an expensive method. examples- forward selection, backward elimination, and recursive FS.
- *Embedded methods:* It is a combination of the filter and the wrapper method. This method is adopted by those are the algorithms that have their built-in FS methods. LASSO regression and RIDGE regression are such examples. The feature's 'usefulness' is measured. The less useful features are removed. The result of this method is a subset of relevant features.

In wrapper methods, we choose a subset of features and the model is trained using them. The information gathered from the previous model is then used to decide whether to add or remove features from the subset. This method is computationally very expensive compared to the other two methods.

- *Forward selection:* Forward selection is an iterative method. In the beginning, the model doesn't contain any features. After every iteration, features are added to the model only until it is improving the performance of the model. The process is stopped when the new variable that is being added starts to reduce the model's performance. Consequently, we get a subset of relevant features.
- *Backward elimination:* It works in the opposite manner of Forward selection method. In the beginning, we have all the features and after every iteration features are removed. The process is only stopped when on removing the feature the performance of the model is reducing. In the end, we are left with only the important features of the dataset (Pathak, 2018).
- *Recursive feature elimination:* It is a greedy optimisation algorithm. Each feature is provided with rank after training the model. This step is repeated multiple times after removing features in every step. The model is trained after every iteration and it remembers the best or the worst performing features. The process is stopped only when no features are left for evaluation. In summary, the features are ranked on the order of their elimination while training the models.

2 Boruta algorithm

Boruta package is most widely used for FS. It uses the wrapper method for implementation. It calculates the importance of a feature with the help of shadow features. The shadow features contain randomly mixed values and are copies of the original features (Kursa and Rudnicki, 2010). The Boruta algorithm is a wrapper built around the random forest classification algorithm. Random forest gives out the importance scores based on Z-score. Z-score alone cannot tell us accurately about the relevance of a feature. We need some other criteria as well to make a distinction between important and unimportant features with respect to the dependent variable. This is where the Boruta algorithm is needed. It tries to collect all the interesting and relevant features (w.r.t output variable) that are present in our dataset.

It works in the following steps:

- Firstly, it creates shadow features also known as permuted copies which are the randomly fixed values and are copies of original features. These features are added to the dataset.
- The model is then trained using all these features. The feature importance measure (commonly used mean decrease accuracy) is then calculated which tells us about the importance of each feature. Higher its value more is its importance.
- The Z-score value is evaluated. At every iteration, it checks whether a real feature has higher importance than the best of its shadow features. This is decided based on whether the feature has a higher Z-score than the maximum Z-score of its shadow features. The unimportant features are removed from the dataset which is reducing the performance of the model.

- Consequently, we are left with important and rejected labeled features. If there are still some tentative features left in the end we can increase the random forest runs which are specified (Pathak, 2018).

3 Research methodology

This research paper aims to estimate the performance of naïve Bayes and decision tree classifier by using a different set of features (Baradwaj and Pal, 2011; Khan, 2005). It is focused on determining the subset of important features. The model is initially trained using all the features of the dataset and the results are recorded. Then, the Boruta algorithm is used to classify the features as confirmed important and rejected. The model is then trained using only those features marked as important and the results are recorded. The results containing accuracy, Precision, Recall, and F-measure are compared in the final result of the two classifiers used.

3.1 Data for the study

The dataset used for this study consists of data on 721 students with 24 exploratory variables. The missing values for quantitative variables are replaced using the average value of that variable and missing values for qualitative variables are removed from the data. This dataset was used in our previous study as well.

3.2 Feature selection

Boruta package available in R software is used for FS in this study. It uses a wrapper algorithm and by default uses random forest. This analysis performed 80 iterations in a total of about 16.46 seconds. The DoTrace is set to 2 so that it reports the decision about the attribute and its importance source run as soon it is concluded.

Figure 1 Boruta result plot for given (educational) data (see online version for colours)

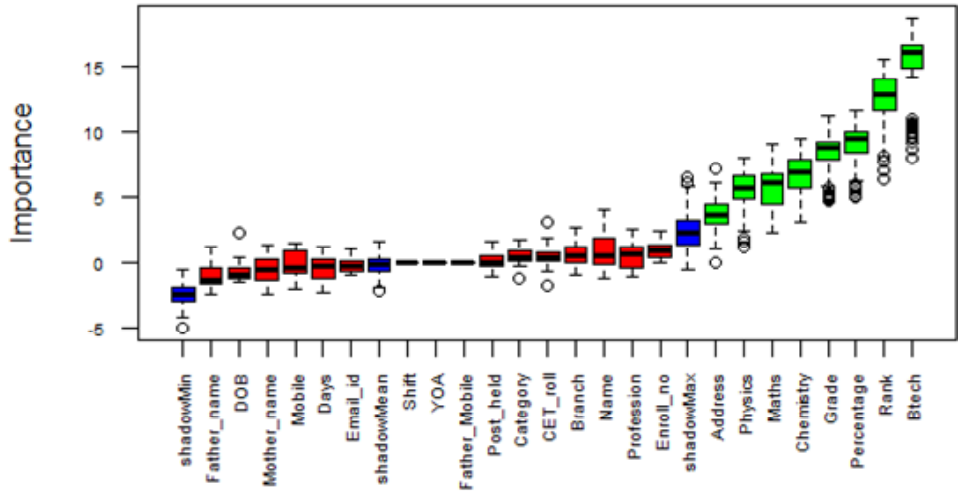


Figure 2 Z score evolution during Boruta run (see online version for colours)

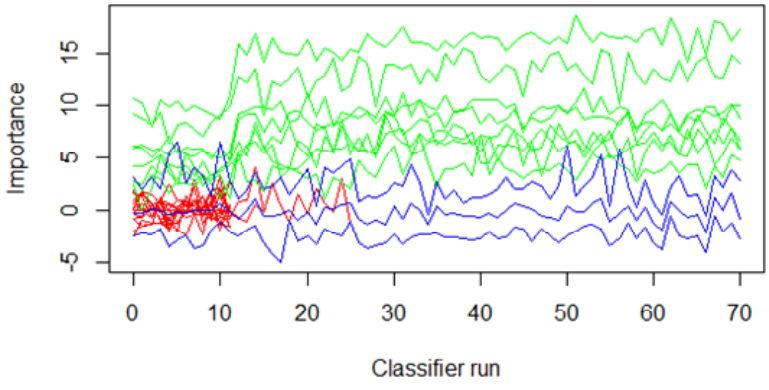


Table 1 Attribute statistics generated by ATTSTATS function

	meanImp	*medianImp*	*minImp*	*maxImp*	*normHits*	*Decision*
Enroll_no	1.053773	0.977365	0.042437	2.458548	0	Rejected
Name	0.790717	0.588191	–1.11638	4.115976	0.056338	Rejected
Category	0.533492	0.477829	–1.15356	1.795627	0	Rejected
Branch	0.686423	0.555963	–0.86785	2.705353	0	Rejected
Shift	0	0	0	0	0	Rejected
YOA	0	0	0	0	0	Rejected
Email_id	–0.1544	–0.15895	–0.87508	1.109782	0	Rejected
Mobile	–0.10712	–0.39838	–1.9597	1.474926	0	Rejected
DOB	–0.57682	–0.96792	–1.48476	2.369705	0	Rejected
CET_roll	0.559263	0.503247	–1.72186	3.118564	0	Rejected
Rank	12.4126	12.88047	6.389109	15.52738	1	Confirmed
Father_name	–1.08215	–1.31938	–2.44054	1.254546	0	Rejected
Father_Mobile	0	0	0	0	0	Rejected
Profession	0.571376	0.757468	–1.05678	2.557946	0	Rejected
Post_held	0.250033	0.08437	–1.03564	1.72883	0.014085	Rejected
Mother_name	–0.53789	–0.56016	–2.448	1.439369	0	Rejected
Address	3.732682	3.740704	0.093973	7.246445	0.704225	Confirmed
Percentage (12^{th})	9.017106	9.518227	5.050247	11.61337	0.971831	Confirmed
Physics	5.523969	5.784632	1.215266	8.006985	0.901408	Confirmed
Chemistry	6.732979	6.912583	3.220916	9.490822	0.943662	Confirmed
Maths	5.77797	6.115813	2.406121	9.118827	0.915493	Confirmed
Days	–0.37856	–0.16819	–2.18907	1.287955	0	Rejected
Btech	15.18032	16.08028	8.042985	18.67254	1	Confirmed
Grade	8.326329	8.802287	4.827359	11.26085	0.957746	Confirmed

Using tentative rough fix, a final classification of 24 features into 8 as important and 16 as unimportant is arrived at (Gopal and Bhargavi, 2018). In Figure 1, boxplots that are green in colour represent features classified as important and red boxplots represent unimportant features. The blue boxplots correspond to the minimum, average and maximum Z-score values of the shadow features. The top three features based on the analysis are Btech percentage, Rank and 12th percentage based on maximum importance values of 18.67, 15.52 and 11.61 respectively.

Figure 2 shows the Z score evolution during Boruta run. Green lines represent the confirmed attributes, red to rejected attributes. The blue lines represent the minimal, average and maximal shadow attribute importance.

Feature groupings based on Boruta analysis are summarised in Table 1. AttStats function is used to show the summary of the Boruta run in the form of a data frame. It contains various importance stats- meanImp, medianImp, etc. as well as the number of the hits that attribute score.

4 Results and discussion

This paper focuses on the relevance of important features in the educational dataset. The success of the classifiers is determined by Precision, Recall, Precision, and Accuracy. FS helps in improving the prediction models used for educational datasets. This process of selecting the relevant attributes helps in creating an accurate predictive model. The academic performance of students plays a pivotal role in their overall development and the need for betterment if their performance is fulfilled by EDM. Predicting the performance of the students helps the educational institutes to guide their students as well as helps in making strategies accordingly. Therefore, FS is used to improve the accuracy of the classifiers (Ramaswami and Bhaskaran, 2001).

Classification algorithms are used in this paper as it is one of the most popular and preferred technique for accurately classifying and predicting the binary variables.

Both of the classification algorithms-naïve Bayes and decision tree were first trained using the training data which contained 70% of the dataset. The remaining 30% was used to test the model. The results were obtained and recorded using the confusion matrix (Elakia and Aarthi, 2014; Tair and El-Halees, 2012; Dekker et al., 2009).

4.1 Classification results using all features

Initially, both the classifiers were applied to all the dataset features and the results are shown in Table 2.

Table 2 Classification results by using all attributes

Evaluation measure	*Naïve Bayes(NB)*	*Decision tree (DT)*
Accuracy	43.59	83.87
Recall	38.09	99.03
Precision	82.75	84.42
F-Measure	52.05	45.56

4.2 Classification using important features:

Further, we selected only the important features (address, physics, maths, chemistry, grade, percentage (12^{th}), rank, Btech) and the unimportant features were dropped. The results are demonstrated in Table 3.

Table 3 Classification results by using important attributes

Evaluation measure	*Naïve Bayes (NB)*	*Decision tree (DT)*
Accuracy	83.89	85.87
Recall	97.38	97.39
Precision	85.63	87.50
F-Measure	91.12	92.18

Figure 3 Result of Naïve Bayes (see online version for colours)

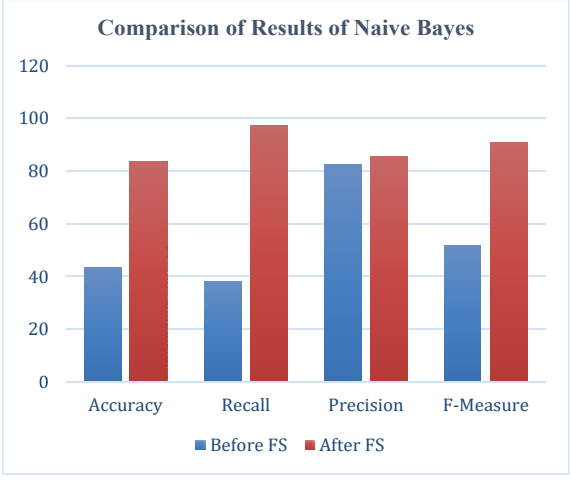
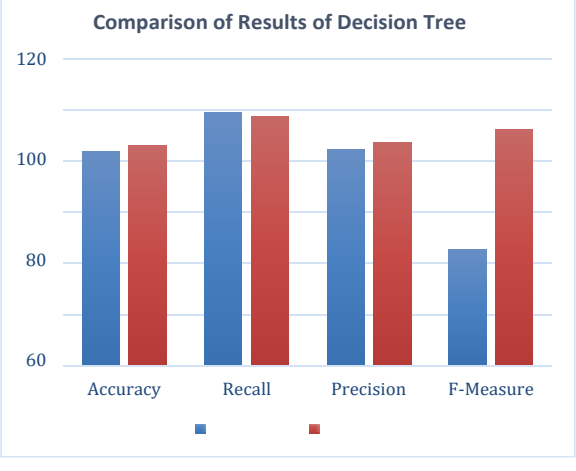


Figure 4 Result of decision tree (see online version for colours)



From results in Tables 2 and 3, it is concluded that the DT classifier has the highest accuracy value. This high accuracy DT is due to its tree hierarchy structure and also, because of the medium size of the dataset used.

The graphical representation of the result of Naïve Bayes is given in Figure 3.

The graphical representation of the result of the decision tree is given in Figure 4.

5 Conclusions

In this study, a FS approach involving the Boruta algorithm is illustrated using educational data. This paper has presented mining of real dataset of college students by using DM classification techniques to predict the performance of students. It has been concluded that Address, physics, maths, chemistry, grade, percentage (12^{th}), rank and Btech are important features that contribute to the performance of students. These features are used for finding accuracy. After applying two classifiers (Naïve Bayes and Decision Tree), it is found that DT classifier gives the best results and achieved an accuracy of 85.87% when used with student's data. The results can further be improved by employing other machine learning approaches and collecting more data.

References

Baradwaj, B.K. and Pal, S. (2011) 'Mining educational data to analyze students' performance', *(IJACSA) International Journal of Advanced Computer Science and Applications*, Vol. 2, No. 6, pp.63–69.

Dekker, G.W., Pechenizkiy, M. and Vleeshouwers, J.M. (2009) 'Educational data mining – EDM', *Proceedings of the 2nd International Conference on Educational Data Mining*, Cordoba, Spain, July, pp.41–50.

Elakia, G. and Aarthi, N.J. (2014) 'Application of data mining in educational database for predicting behavioural patterns of the students', *(IJCSIT) International Journal of Computer Science and Information Technologies*, Vol. 5, No. 3, pp.4649–4652.

Gopal, M. and Bhargavi, P.S. (2018) 'Feature selection for yield production using Boruta algorithm', *International Journal of Pure and Applied Mathematics*, Vol. 118, No. 22, pp.139–144.

Khan, Z.N. (2005) 'Scholastic achievement of higher secondary students in science stream', *Journal of Social Sciences*, Vol. 1, No. 2, pp. 84–87.

Kursa, M.B. and Rudnicki, W.R. (2010) 'Feature selection with the Boruta package', *Journal of Statistical Software*, September, Vol. 36, No. 11, pp.1–10.

Pathak, M. (2018) *Feature Selection in R*, 7 March, Datacamp [online] https://www.datacamp.com/community/tutorials/feature-selection-R-boruta#boruta (accessed 21 June 2019).

Ramaswami, M. and Bhaskaran, R. (2001) 'A study on feature selection techniques in educational data mining', *Journal of Computing*, December, Vol. 1, No. 1, pp.7–11.

Saurav, K. (2016) 'Introduction to feature selection methods with an example', *Analytics Vidya*, 1 December [online] https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example- or-how-to-select-the-right-variables/ (accessed 10 July 2019).

Tair, M.M.A. and El-Halees, A.M. (2012) 'Mining educational data to improve student's performance: a case study', *International Journal of Information and Communication Technology Research*, Vol. 2, No. 2, pp.140–146.

Velmurugan, T. and Anuradha, C. (2016) 'Performance evaluation of feature selection algorithms in educational data mining', *International Journal of Data Mining Techniques and Applications*, 2 December, Vol. 5, pp.131–139.

Zaffar, M., Savita, K.S., Hashmani, M.A. and Rizvi, S.S.H. (2018) 'A study of feature selection algorithms for predicting students academic performance', *(IJACSA) International Journal of Advanced Computer Science and Applications*, Vol. 9, No. 5, pp.541–549.