Date of publication JUN-30, 2022, date of current version MAR-21, 2022.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.21.2.2589

Examining Techniques to Solving Imbalanced Datasets in Educational Data Mining Systems

AHMED AL-ASHOOR[1], SHUBAIR ABDULLAH[2]
[1]University of Basra, Basra, Iraq, ahmed.ashur@uobasrah.edu.iq
[2]Sultan Qaboos University, Muscat, Oman, shubair@squ.edu.om

Corresponding author: Ahmed Al-Ashoor (e-mail: ahmed89.alashoor89@gmail.com).

ABSTRACT The educational data mining research attempts have contributed in developing policies to improve student learning in different levels of educational institutions. One of the common challenges to building accurate classification and prediction systems is the imbalanced distribution of classes in the data collected. This study investigates data-level techniques and algorithm-level techniques. Six classifiers from each technique are used to explore their effectiveness to handle the imbalanced data problem while predicting students' graduation grade based on their performance at the first stage. The classifiers are tested using the k-fold cross-validation approach before and after applying the data-level and algorithm-level techniques. For the purpose of evaluation, various evaluation metrics have been used such as accuracy, precision, recall, and f1-score. The results showed that the classifiers do not perform well with imbalanced dataset, and the performance could be improved by using these techniques. As for the level of improvement, it varies from one technique to another. Additionally, the results of the statistical hypothesis testing confirmed that there were no statistically significant differences for classifiers of the two techniques.

KEYWORDS educational data mining; machine learning; imbalanced datasets; prediction; student grade.

I. INTRODUCTION

THE Educational Data Mining (EDM) uses Data Mining (DM) and Machine Learning (ML) techniques to extract the knowledge from hidden information and to predict the main factors that play a significant role in students' performance [1]. The main idea of an EDM technique is based on collecting data about the students, such as personal information, academic information, and test scores, and building models to predict students' academic performance.

A potential goal of an EDM technique is to better identify the settings needed to improve students' outcomes, to understand students' behavior, to improve teaching process, to improve e-learning systems, and to identify reasons for dropping out [2]. Predicting student performance can help determine academic advising needs at an early stage, thus retaining students until degree completion, which is one of the significant challenges facing academic institutions. Moreover, this kind of prediction can give a proper warning to the students at risk and thus, help them to overcome the difficulties of studying [3]. However, predicting students' performance depends on various factors or characteristics, such as personal characteristics, demographics, environmental factors, and academic progress, which makes it quite challenging [4]. The overlap between the values of these factors and their variability due to human nature have made the prediction process more complicated and created challenges preventing development of high-precision models in some settings. The field of EDM seeks ways to find interesting information in the data. Beside the challenge of querying and processing huge amounts of information quickly and accurately, there is another challenge that is represented by dealing with small amount of data and class imbalance distribution. To deal with these challenges, researchers are exploring new methods that are studied in the computation theory field such as randomization [5].

Many EDM approaches that employ machine learning techniques to discover meaningful patterns for predicting student's performance have been introduced recently. The ML is subfield of artificial intelligence that aims at extracting knowledge from data. The data used in developing ML models is divided into two groups: training data and testing data. Training data is used to train an ML algorithm to accurately predict certain outputs while testing data is used to measure the performance of the trained ML algorithms. According to the method followed during the training phase, ML algorithms are divided into three types, supervised, unsupervised, and semi-supervised. In supervised algorithms, the training data consists of features and labels, while in unsupervised algorithms, the training data consists of features. The semi-supervised algorithms use combination of supervised and unsupervised training data [6].

This study tries to contribute in overcoming the imbalance distribution challenge in educational dataset by investigating the techniques that deal with the imbalanced dataset problem. The techniques addressed in this study are data-level technique and algorithm-level technique. These two techniques have been widely used to overcome the problem of building an accurate classifiers using imbalanced datasets. A data-level technique attempts to balance the classes in the training data before training a classifier, whereas an algorithm-level technique attempts to modify the classifier to make it appropriate for imbalanced datasets. This study examines whether these two techniques have an effect on the classifiers' accuracy in the EDM systems. Another contribution is to determine whether the use of random methods that form important research filed of theory of computing in oversampling is feasible. The data used in this study contains information about the students of the College of Pharmacy in the University of Basra. It includes students' degrees in six courses of the first stage in the Bachelor Program, Mathematics and Biostatics, Human Biology, Analytical Chemistry, Principles of Pharmacy Practice, Human Rights, and Computer Basics.

The remainder of this paper is organized as follows: Section 2 reviews the literature. The research methodology is explained in Section 3. Sections 4 provides the result and discussion of experiments conducted. And finally, Section 5 lists the findings and concludes the research.

II. LITERATURE REVIEW

The imbalanced data refers to non-equal representation of classes in datasets. Unfortunately, it is almost the dominant feature of datasets of EDM systems as these systems usually depend on dataset with values related to human nature like student's marital status, economic status, the number of family members, the number of study hours, the number of sleeping hours, etc. Due to this type of value, the dataset does not have an equal number of instances in each class. A small difference between the number of classes often does not matter, but this not the usual case. For example, the datasets that characterize student's graduation grades are highly imbalanced, e.g., the majority of the students will be in the "normal AGPA" or "high AGPA" class and small minority will be in the "low AGPA" class [7].

Many techniques have been proposed to handle the imbalanced dataset problem. These techniques could mainly be categorized into two groups: data-level and algorithm-level. The data-level techniques aim at resampling the datasets by either increasing or decreasing the frequency of samples in classes. These techniques are simplest and effective techniques to handle the imbalanced datasets problem. Mainly, they are divided into three approaches, undersampling, oversampling, and hybrid. The undersampling approach is the process of eliminating some majority class samples. The oversampling approach is the opposite; it involves synthesizing new minority class samples. The hybrid approach combines the two methods to achieve the balanced class distribution [8]. An algorithm-level technique, which might be called ensemble-based classifier, focuses on improving classifiers rather than the datasets. The main idea of these techniques is to adopt a special strategy of merging several classifiers from one original dataset into one classifier, and then aggregating the classification results. These algorithm-level techniques have been extensively adopted to handle the imbalanced dataset problem [9], and there are several approaches proposed to build ensemble classifiers [10]. Examples of advanced proposed approaches include Stacking, Bagging, and Boosting. In the stacking, multiple classifiers are trained on a single dataset. During the test phase, all samples from the testing data are classified by all classifiers. The results of training and testing the classifiers compose the training dataset for a resultant model called meta-model. Fundamentally, bagging approach works by choosing samples from the original training dataset into subsets of samples and fitting a decision tree on each subset. Each decision tree will make a classification based on its subset of samples, and the final classification is made by combining the results of all decision trees using simple statistics such as averaging. The boosting approach works similar to bagging approach. The difference is that all decision trees use the same original training dataset, which means no sampling process for the original training dataset is carried out. Another difference between boosting and bagging is that the bagging approach assigns a weight for items in the dataset during the training phase [11].

Despite the large size of the challenges raised by the problem of imbalance samples distribution in EDM, the published research did not address these challenges in a way that reflects their size [12]. However, there are several works published to compare different classifiers. In [13], various machine learning algorithms were compared to predict students' performance and show that the decision tree has the ability to perform at high level. Similarly, the work published by [14] aimed at assessing the performance of different classifiers with a focus on feature selection method. Our study is similar to these studies in that it compares a number of classifiers, but it differs in that the comparison aims to study the problem of dataset imbalance. The work that might be considered close to this study is published by [12]; however, their study aimed to investigate the problem of imbalance problem by comparing the data resampling methods.

The past two decades have witnessed an increase and diversity of published research on the use of DM and ML technologies to predict students' academic performance. The published researches addressed an important topic such as students' modelling, and recommendations for future planning [15]. [16] introduced a case study in the Open Polytechnic of New Zealand. There was applied a Classification and Regression Tree (CART) on a dataset consisted of 450 students to investigate the impact of using some enrolment data, non-academic attributes to predict students' success. The study concluded that students' culture or ethnicity is one of the main factors affecting students' performance. Tsai et al. [17] applied the K-means algorithm, unsupervised neural networks, and the C5.0 decision-tree algorithm at the National University in Taiwan to cluster and predict the undergraduate students. The purpose of the study was to develop an early-warning system to identify the students who might fail one of the graduation requirement tests, the computer proficiency. Based on their test results, the authors recommended the K-means algorithm as the most effective. The fuzzy inference models are applied as well. The work of [18] developed a classification model to classify students who may graduate with a low GPA as at Sultan Qaboos University, in Oman. The authors in [19] examined the effectiveness of two methods for semi-supervised learning in predicting students' final examination scores in high school. They evaluated the performance of self-training and Yet Another Two Stage Idea (YATSI) approaches. The selected attributes were related to written assignments, oral examinations, short tests, and examinations. Based on their numerical experiments, the student's classification accuracy of these approaches can be significantly improved.

III. METHODOLOGY

A. DATASET

The data used in this study has been provided by Basra University. The dataset includes list of undergraduate students from the period 2011-2015. A set of 536 student records have been extracted from the database of the examination committee in the College of Pharmacy. In order to predict the graduate grade at an early stage of the study program, we collected students' degrees in five core courses of the first stage, Mathematics and Biostatics, Human Biology, Analytical Chemistry, Principles of Pharmacy Practice, Human Rights, and Computer Basics, along with the graduation grade of the students. Table 1 shows a description of the datasets provided.

Table 1. Dataset used in this study

Cohort	Male	Female	Total
2011-2012	85	96	181
2012-2013	50	53	103
2013-2014	61	69	130
2014-2015	56	66	122
Total	252	284	536

The preprocessing stage involved only converting the categorical values of graduation grades into numerical values. This step is important to setup the inputs of machine learning models. Table 2 shows the graduation grades categorical representations and their equivalent values after conversion. The "fail" class was excluded since it has very few samples.

Table 2. Categorical and numerical representations of graduation grades

Grade Range	Categorical representation	Numerical value
100-90	excellent	5
89-80	very-good	4
79-70	good	3
69-60	average	2
59-50	accept	1
Less than 50	Fail	0

Regarding the data representation, given a set of input degrees X ⊑ R {0..100} and x1, x2, x3, x4, x5, x6 ∈ X, where x1 ... x6 represent the student's degrees in five core courses. Y={1,2,3,4,5} is the list of graduation grades and y ∈ Y represents the student's graduation grade. A classifier f(x) is learned from a given set of (x1, x2, x3, x4, x5, x6, y). Table 3 shows some samples of data.

The analysis of dataset reveals that distribution of the six classes of students based on their graduation grades are imbalanced. The classes Very-good and Excellent include low number of samples (1% of samples in Very-good class and 2% of samples in Excellent class), while the other two classes have the majority of samples (39% of samples in Good class and 42% of samples in Average class). Fig. 1 shows the distribution of the students based on their graduation grades in the introduced datasets.

Table 3. Data sample

Student #	Math. & Biostatics (x_1)	Human Biology (x_2)	Analytical Chemistry(x_3)	Prin. of Phar. Practice (x_4)	Human Rights (x_5)	Computer Basics (x_6)	Graduation grade (y)
1	76	52	82	61	79	62	2
2	77	76	63	56	61	65	2
3	67	50	82	68	65	55	4
4	56	50	74	50	50	50	1
5	50	50	82	50	50	53	2
6	57	55	69	50	58	51	1
7	54	52	76	51	55	63	5
8	57	58	77	51	52	52	1
9	50	59	69	51	55	63	2
10	54	50	69	60	50	53	3

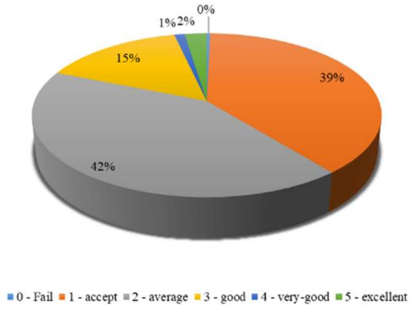

Figure 1. Distribution of the students based on their graduation grades in the datasets.

B. DATA IMBALANCE HANDLING TECHNIQUES

We focus on the two techniques, data-level and algorithm-level. Different classifiers were used to test each technique. With regard to the data-level, we focused on the Support Vector Machine – Synthetic Minority Over-Sampling Technique (SVM-SMOTE) and the Random Undersampling (RUS) methods. The SMOTE method is used to synthesize new minority class samples using the SVM model whereas the RUS method is used to trim samples of the majority class randomly. These two methods were selected due to their emergence in literature published recently as one of the best methods to deal with imbalanced datasets of EDM [12]. The classifiers along with SMOTE and RUS are listed in Table 4, which also shows the setup of each of them.

Table 4. Classifiers used in this study

Classifier	Setup	Ref.
K-nearest-neighbor (kNN)	n_neighbors=3	[20]
Support Vector Machine (SVM)	kernel='linear'	[21]
Multilayer perceptron (MLP)	hidden layer sizes=(5, 2)	[22]
Decision Trees (DT)	Default	[23]
Naïve Bayes (NB)	Deault	[24]
Logistic Regression (LR)	multi_class='multinom'	[25]

Regarding the algorithm-level techniques, we focused on different ensemble classifiers from the staking, bagging, and boosting approaches. The ensemble classifiers used in this paper are listed in Table 5, which also shows the setup of each classifiers and the classifiers' approaches.

Table 5. Ensemble classifiers used in this study

Classifier	Setup	Ref.
Bagging (Bg)	n_estimators=50	[26]
Gradient Boosting (GB)	n_estimators=50	[26]
Adaptive Boosting (AdB)	n_estimators=50	[27]
Random Forest (RF)	Default	[28]
Stacking (St)	Default	[11]
Extreme Gradient Boosting (XGB)	Default	[26]

C. VALIDATION AND EVALUATION

In this paper, we applied the k-fold cross-validation, which is a validation model applied to evaluate the effectiveness of machine learning algorithms. One main advantage of k-fold cross-validation is that all samples are used for both training and testing the algorithm and every single sample being used for validation exactly once [29]. We randomly divided the dataset into 90% of samples for training and 10% of samples for testing and evaluated the algorithm. For the purpose of evaluation, various evaluation metrics have been used such as accuracy, precision, recall, and f1-score. The random division and evaluation were repeated 10 times to get the algorithm trained and evaluated on the entire dataset. The data-level and algorithm-level approaches were applied to the training datasets only while the testing datasets were not balanced during the validation and evaluation. Since we focus on two techniques and six classifiers from each technique, there will be sets of results showing the performance of the classifiers resulting. In this case, it is difficult to analyze and compare the results using accuracy, precision, recall, and f1-score evaluation metrics. Therefore, we used hypothesis statistical tests to solve this problem.

IV. RESULTS AND DISCUSSION

The main goal of this study is to deal with the imbalance challenges in datasets in educational data mining systems. Since the study attempts to tests two techniques with six classifiers from each technique, several test are conducted. All the experiments were carried out on a PC with 3.8 GHz Intel Core i5 CPU and 8 GB of RAM, and the classifiers have been developed in Python. This section explains the experiments and also shows and discusses the results.

A. IMBALANCE PROBLEM IN EDM SYSTEMS

The tests in this section aims to show the impact of the problem of imbalanced data on the performance of classifiers in EDM system. The experiments were conducted to measure the accuracy of the classifiers in classifying students according to their graduation grades. Table 6 shows the results of the tests by displaying the values of the measures of accuracy, precision, recall, and F1-score. All the tests were carried out using the technique of cross validation to guarantee testing classifiers on unseen samples. The accuracy is the proportion between the number of correct classifications and the total number of samples tested. When applying the k-fold cross-validation, which means splitting the dataset to a number of bunches of train/test samples, the test accuracy is calculated from each bunch and then the results are averaged together. Five classifiers performed at the accuracy below 60%, which reveals that most of classifiers have not achieved satisfactory results. The MLP classifier achieved the lowest accuracy among all other classifiers 45%. The DT classifiers performed at 95% accuracy, which reflects an excellent performance on the dataset. The reason for this high accuracy could be attributed to the fact that the dataset being used in the study contains multiple outliers and peaks. Fig. 2 graphically offers information about the outliers in the dataset values, where we can see in most courses several outliers located outside the interquartile range (IQR). Since the DT classifier is preferred for non-parametric data [30, 31], the high values of testing accuracy were recorded.

Despite the importance of accuracy metric as indication of classifiers' performance, the precision and recall metrics should be considered as well. Similar to accuracy, the precision and recall were calculated from each bunch of train/test samples and then the results were averaged together. In classification evaluation, the precision metric quantifies the number of correctly classified samples that actually belong to their classes. The precision metric for the MLP classifier was approximately 28%, which is the lowest precision among the classifiers. The remarkable precision value, 96% was recorded for the DT classifier, and the high precision was recorded for all classes, for example 96% of students that classified both of the classifiers as "average" are actually "average". The recall metric quantifies the number of correctly classified samples in every class made out of all sample in every class. The lowest recall value 45% was recorded for the MLP classifier. The recall values of MLP classifier for "accept" class and "average" class are 45% and 46% respectively of actual samples in these two classes. On the other hand, the remarkable result was 95% recorded for DT classifier. It is worth saying that the precision and recall values of DT classifier reinforce the hypothesis that this classifier performs very well in the dataset used.

The F1-score provides a single harmonic average of precision and recall that balances their values in one number. Apart from the DT classifier, the F1-score values recorded with each class reveal low performance of the classifiers. The kNN, SVM, MLP, NB, and LR achieved below 60% F1-score values. Accordingly, the performance of these classifiers is not acceptable. The main reason for this low performance in the majority of classifiers tested is the imbalance problem in the dataset used in the study.

To further examine and investigate the impact of imbalanced dataset problem on the performance of classifiers, the classification problem was transformed from multi-class to binary classification. This step helps in measuring the impact of imbalanced dataset on many binary classification EDM systems such as the systems that attempt to classify students based on results (pass or fail) or based on engagement in online learning (engaged or not engaged). Accordingly, the five classes (accept, average, good, very-good, and excellent) that represent graduation grades for students have been reconfigured and reduced to two classes, "high" and "low". The classes "accept" and "average" are included in "low" class, and the classes "good", "very-good" and "excellent" are included in "high" class.

Table 6. Performance of classifiers on the imbalanced dataset

	Accuracy (%)						Precision (%)					
	accept	average	good	v. good	excellent	avrg	accept	average	good	v. good	excellent	avrg
kNN	47	46	48	47	46	46.8	48	47	46	49	48	47.6
SVM	58	57	57	59	57	57.6	49	52	49	51	48	49.8
MLP	45	46	45	45	44	45	26	28	27	28	29	27.6
DT	96	97	94	96	94	95.4	96	96	96	97	95	96
NB	59	58	57	57	57	57.6	58	59	60	56	60	58.6
LR	47	47	49	47	74	52.8	39	41	43	42	42	41.4

	Recall (%)						F1-score (%)					
	accept	average	good	v. good	excellent	avrg	accept	average	good	v. good	excellent	avrg
kNN	50	51	46	45	44	47.2	51	46	50	44	46	47.4
SVM	57	59	60	61	60	59.4	56	54	54	53	57	54.8
MLP	45	46	47	43	44	45	28	28	28	29	26	27.8
DT	95	95	95	96	94	95	95	95	95	96	96	95.4
NB	59	58	58	57	57	57.8	57	55	58	57	57	56.8
LR	48	46	49	47	48	47.6	45	45	44	43	43	44

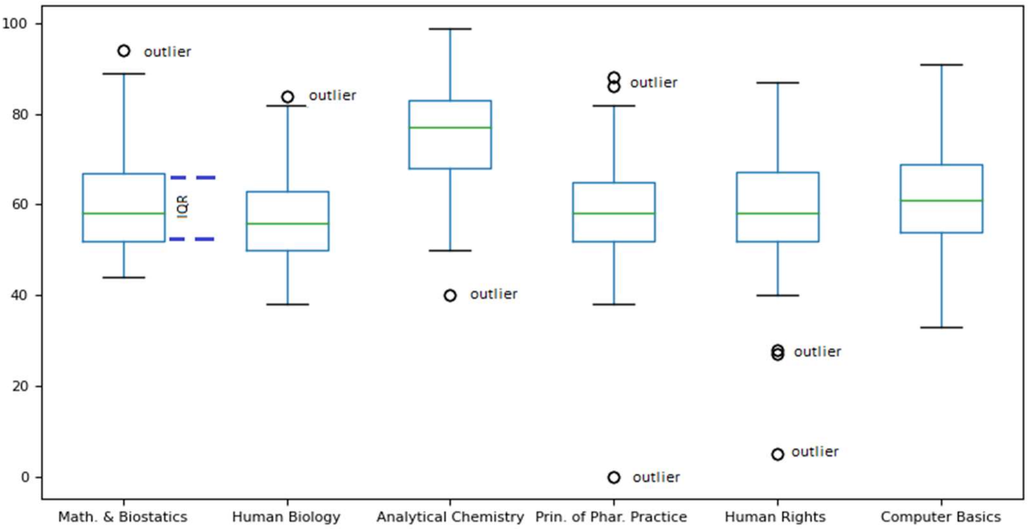

Figure 2. Outliers in the dataset.

Despite that the reconfiguring process has transformed the dataset from multi-class problem to binary classification problem, it has also produced a highly imbalanced dataset. Fig. 3 shows distribution of the students after reducing the six classes into two classes. Table 7 shows results of testing performance of the classifiers in classifying the students into two classes, binary classification. The experiments were conducted using the cross validation technique.

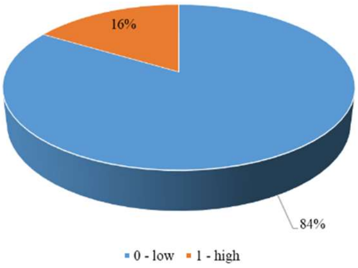

Figure 3. Distribution of the students into "low" and "high" classes.

Table 7. Performance of Classifiers on the imbalanced two-class dataset

	Accuracy			Precision		
	pass	fail	average	pass	fail	average
kNN	75	75	75	76	77	76.5
SVM	85	83	84	71	70	70.5
MLP	83	84	83.5	71	70	70.5
DT	96	97	96.5	97	97	97
NB	80	80	80	83	83	83
LR	84	82	83	80	79	79.5

	Recall			F1-score		
	pass	fail	average	pass	fail	average
kNN	75	75	75	75	75	75
SVM	83	84	83.5	75	77	76
MLP	84	84	84	76	75	75.5
DT	98	97	97.5	97	97	97
NB	80	79	79.5	80	81	80.5
LR	82	84	83	79	80	79.5

The results of binary classification tests show that the performance of majority of classifiers (5 out of 6) has clearly improved. The accuracy of kNN, SVM, MLP, NB, and LR did not exceed 84%, which indicates a margin for enhancement. The overall results show that reducing number of classes to only two might reduce but not eliminate the impact of imbalance class distributions on the performance of the classifiers.

B. TESTING DATA-LEVEL TECHNIQUES

According to [32], the hybrid approach that involves implementing combination of SMOTE and RUS performs better than choosing one approach. We followed first a resampling procedure to trim the number of samples in the majority classes ("accept" class, "average" class, and "good" class) using the RUS method, then we created synthetic samples from the minority classes ("very-good" class and "excellent" class). Fig. 4 shows scatter plot of samples by class label before and after implementing the SMOTE and RUS methods. In addition to get balanced number of classes in the dataset, the resampling was aimed also at obtaining the normal distribution of students' grades, in which the data near the mean are more frequent in occurrence than far from the mean. Fig. 5 shows distributions of classes before and after implementing SMOTE and RUS methods. The change during resampling procedures involved increasing or decreasing number of classes without changing the total number of 536 samples in the dataset.

Table 8 shows the results of testing the performance of classifiers after implementing the SMOTE and RUS resampling methods. The performance is quantified by the values of accuracy, precision, recall, and F1-score. Similar to the previous experiments, the technique of cross validation is used to guarantee testing classifiers on unseen samples.

In contrast with the test results before implementing SMOTE and RUS methods, the performance of majority of classifiers is either almost the same as in SVM, MLP, NB, and LR or has recorded improvements as in kNN. The notable result is associated with the DT classifier, where we find that its performance decreased by approximately 15%. The accuracy of DT before implementing SMOTE and RUS methods was 95%, and after implementing these methods, the accuracy decreased to 81%. The decrease in DT performance can be attributed to the fact that this algorithm is largely unstable compared to other classifiers. Small change in the data might cause restructure of tree, which lead to different performance from what is expected in normal event [33]. The improvement in the performance of kNN cannot be generalized over all classifiers as it occurred only for the kNN algorithm m.

C. TESTING ALGORITHM-LEVEL TECHNIQUES

This section shows the results of testing the algorithm-level techniques. The six ensemble classifiers involved in the tests are Bg, GB, AdB, RF, St, and XGB. Table 9 summarizes the results of testing the ensemble classifiers on the imbalanced original dataset. Four classifiers Bg, RF, St, and XBG achieved accuracy 90% and above approximately. The St ensemble classifier achieved the highest rate of accuracy 94.8%. The precision, recall, and F1-score values of St classifier are almost 90%, which means that the St ensemble classifier outperformed other ensemble classifiers. During the experiments, we combined six classifiers kNN, SVM, MLP, DT, BN, and LR in implementing the St ensemble model, and this might be the reason behind the excellent performance. The Bg and XGB ensemble classifiers could be considered also as their values for all measurements not less than 90% approximately. On the other hand, the AdB classifier has not performed well and achieved the lowest accuracy value 48.2% among all the ensemble classifiers. This low performance is due to the nature of the classifier as it is a special case with a particular loss function and not flexible compared to GB and XGB classifiers, for example [34].

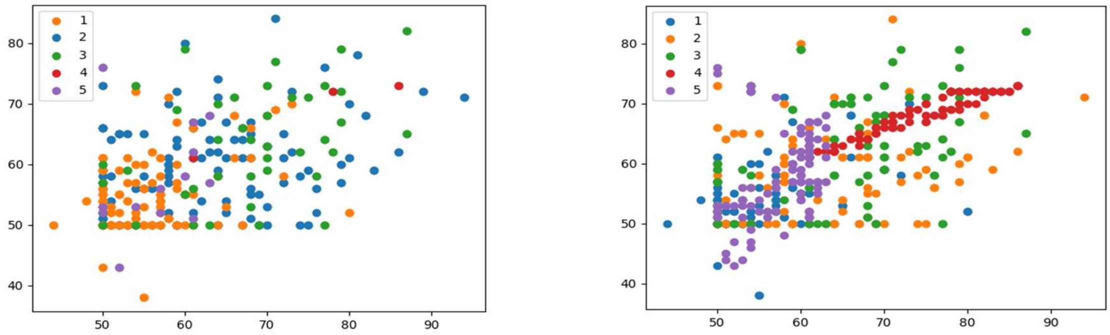

Figure 4. Scatter plot of samples by class label before (left) and after (right) implementing the SMOTE and RUS methods

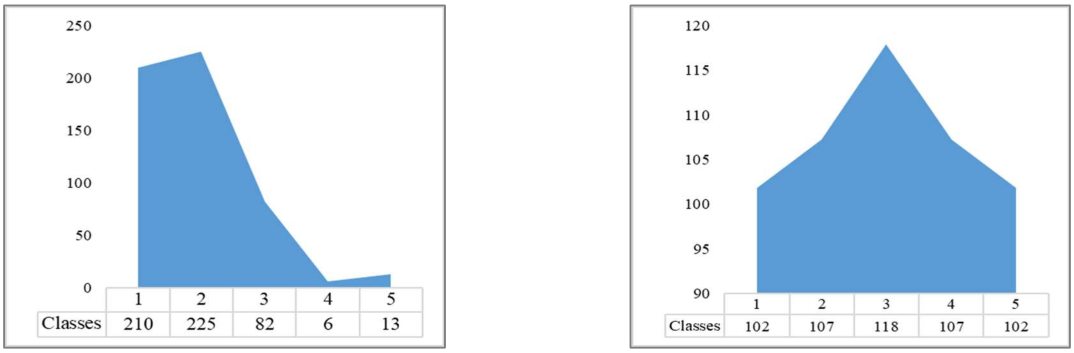

Figure 5. Distributions of classes before (left) and after (right) implementing the SMOTE and RUS methods

Table 8. Performance of classifiers on the balanced dataset

	Accuracy (%)						Precision (%)					
	accept	**average**	**good**	**v. good**	**excellent**	**avrg**	**accept**	**average**	**good**	**v. good**	**excellent**	**avrg**
kNN	81	80	80	81	81	80.60	78	80	81	80	78	79.40
SVM	61	61	61	60	60	60.60	55	55	57	57	57	56.20
MLP	49	50	50	50	50	49.80	25	25	25	24	24	24.60
DT	81	81	81	80	81	80.80	81	81	82	80	80	80.80
NB	55	55	55	55	54	54.80	55	55	56	56	57	55.80
LR	54	54	55	55	55	54.60	38	38	38	36	38	37.60
	Recall (%)						**F1-score (%)**					
	accept	Average	**good**	**v. good**	**excellent**	**avrg**	**accept**	**average**	**good**	**v. good**	**excellent**	**avrg**
kNN	81	81	80	80	81	80.60	78	77	78	77	78	77.60
SVM	60	61	61	61	61	60.80	54	54	56	56	55	55.00
MLP	50	50	50	50	51	50.20	34	34	35	35	35	34.60
DT	80	80	80	81	82	80.60	81	80	79	79	80	79.80
NB	54	54	56	56	55	55.00	52	53	53	50	50	51.60
LR	56	56	55	54	54	55.00	55	56	54	54	56	55.00

Table 9. Performance of ensemble classifiers on the imbalanced dataset

	Accuracy (%)						Precision (%)					
	accept	**average**	**good**	**v. good**	**excellent**	**avrg**	**accept**	**average**	**good**	**v. good**	**excellent**	**avrg**
Bg	91	90	91	91	91	90.8	90	91	91	91	91	90.8
GB	79	77	77	78	78	77.8	79	79	77	78	78	78.2
AdB	50	48	48	48	47	48.2	44	43	43	43	45	43.6
RF	91	90	92	90	91	90.8	91	92	91	92	92	91.6
St	95	95	94	95	95	94.8	90	90	90	91	89	90.0
XGB	89	90	90	90	89	89.6	89	90	91	91	90	90.2
	Recall (%)						**F1-score (%)**					
	accept	**average**	**good**	**v. good**	**excellent**	**avrg**	**accept**	**average**	**good**	**v. good**	**excellent**	**avrg**
Bg	90	91	92	90	89	90.4	91	90	90	91	91	90.6
GB	78	77	77	77	77	77.2	77	78	77	77	78	77.4
AdB	49	50	50	52	49	50	46	46	46	46	46	46
RF	90	90	90	91	90	90.2	91	91	90	90	91	90.6
St	90	90	91	91	89	90.2	90	90	89	90	90	89.8
XGB	90	89	89	90	91	89.8	89	89	90	90	90	89.6

D. STATISTICAL TESTS

The experiments conducted provided two sets of accuracy means showing some enhancement in the performance of six classifiers from each technique. The challenge here is to find the best technique that achieves the most accurate classification. The ANOVA test is considered. To get a trustable result, we checked the normality of the data using the Shapiro-Wilks normality test before applying the ANOVA test. The null-hypothesis of this normality test is that the data is normally distributed. The Shapiro-Wilks normality test resulted in p-value 0.0605 greater than 0.05. Therefore, the null-hypothesis is accepted and the ANOVA test was applied. Table 10 shows the results of Shapiro-Wilks normality test.

Table 10. Results of Shapiro-Wilks normality test

Technique	**Samples**	**Mean**	**Std**	**p-value**
Data-level	6	63.53	12.53	0.0605
Algorithm-level	6	82.00	16.00	

The ANOVA test is applied for comparing the results of the two techniques. The null-hypothesis of ANOVA test is that there is no difference between the two groups of results. The p-value resulted is greater that the significance-level (alpha = 0.05). Therefore, the test failed to reject the null hypothesis, and it can be concluded that the performance of the two methods is statistically equal on the dataset.

V. CONCLUSION

The educational data mining research area has recently got more attention. Attempts to predict student graduation and classify students based on their learning styles have helped educational institutions develop policies to improve student learning. One of the big and common challenges to building accurate classification and prediction systems is the imbalanced distribution of classes in the data collected. This study examined the data-level techniques and algorithm-level techniques that are used to overcome the problem of building an accurate classifiers using imbalanced datasets. Six classifiers from each technique have been selected to conduct the experiments. The classifiers are tested using the k-fold cross-validation approach before and after applying the data-level and algorithm-level techniques. For the purpose of evaluation, various evaluation metrics have been used such as accuracy, precision, recall, and f1-score.

The results of experiments have confirmed the effect of imbalance dataset on the classification accuracy of most of the classifiers tested. However, the DT classifiers showed good performance on the imbalanced dataset used in this study as the dataset contains multiple outliers. This result may encourage using the DT for this type of dataset. The results of tests after transforming the problem to binary classification showed that the performance of majority of classifiers (5 out of 6) has clearly improved on the imbalanced dataset. However, there was a margin to improve the performance, and this reveals that transforming multi classification problem to binary classification problem might reduce the effect of imbalance dataset problem but not eliminate it. Regarding the data-level techniques, the SMOTE and RUS methods have been implemented and six classifiers were tested, and the results have not shown much improvement. This might conclude that the randomization used in the RUS has no strong positive effect in solving the imbalanced classes problem over the education dataset. The kNN classifier achieved high level of performance. This improvement cannot be generalized over all classifiers as it occurred only for the kNN classifier. The results of testing ensemble classifiers show that Bg, RF, St, and XBG from the algorithm-level techniques achieved accuracy 90% and above approximately. This indicates that some ensemble classifiers work much better than data-level classifiers on imbalanced data. Despite slight improvement of data-level classifiers and the significant improvement of the algorithm-level ensemble classifiers, the statistical hypothesis tests showed that there are no significant statistical differences between the two techniques.

References

[1] R. Kamath and R. Kamat, Educational Data Mining with R and Rattle, *River Publishers*, 2016.

[2] C. Romero, S. Ventura, "Educational data mining: a review of the state of the art," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 6, pp. 601-618, 2010. https://doi.org/10.1109/TSMCC.2010.2053532.

[3] B. Guo, R. Zhang, G. Xu, C. Shi, and L. Yang, "Predicting students performance in educational data mining," *Proceedings of the 2015 IEEE International Symposium on Educational Technology (ISET)*, 2015, pp. 125-128, https://doi.org/10.1109/ISET.2015.33.

[4] R. Asif, A. Merceron, S. A. Ali, N. G. Haider, "Analyzing undergraduate students' performance using educational data mining," *Computers & Education*, vol. 113, pp. 177-194, 2017. https://doi.org/10.1016/j.compedu.2017.05.007.

[5] Y.-H. Hu, C.-L. Lo, S.-P. Shih, "Developing early warning systems to predict students' online learning performance," *Computers in Human Behavior,* vol. 36, pp. 469-478, 2014. https://doi.org/10.1016/j.chb.2014.04.002.

[6] M. Alloghani, D. Al-Jumeily, J. Mustafina, A. Hussain, and A. J. Aljaaf, "A systematic review on supervised and unsupervised machine learning algorithms for data science," *Supervised and Unsupervised Learning for Data Science*, pp. 3-21, 2020. https://doi.org/10.1007/978-3-030-22475-2_1.

[7] S. Datta, A. Arputharaj, "An analysis of several machine learning algorithms for imbalanced classes, *Proceedings of the 2018 5th IEEE International Conference on Soft Computing & Machine Intelligence (ISCMI)*, 2018, pp. 22-27. https://doi.org/10.1109/ISCMI.2018.8703244.

[8] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463-484, 2011. https://doi.org/10.1109/TSMCC.2011.2161285.

[9] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Systems with Applications*, vol. 73, pp. 220-239, 2017. https://doi.org/10.1016/j.eswa.2016.12.035.

[10] Z. Sun, Q. Song, X. Zhu, H. Sun, B. Xu, Y. Zhou, "A novel ensemble method for classifying imbalanced data," *Pattern Recognition*, vol. 48, no. 5, pp. 1623-1637, 2015. https://doi.org/10.1016/j.patcog.2014.11.014.

[11] A. Jurek, Y. Bi, S. Wu, C. Nugent, "A survey of commonly used ensemble-based classification techniques," *The Knowledge Engineering Review*, vol. 29, no. 5, p. 551, 2014. https://doi.org/10.1017/S0269888913000155.

[12] R. Ghorbani, R. Ghousi, "Comparing different resampling methods in predicting Students' performance using machine learning techniques," *IEEE Access*, vol. 8, pp. 67899-67911, 2020. https://doi.org/10.1109/ACCESS.2020.2986809.

[13] A. Acharya, D. Sinha, "Early prediction of students performance using machine learning techniques," *International Journal of Computer Applications*, vol. 107, no. 1, pp. 37-43, 2014. https://doi.org/10.5120/18717-9939.

[14] F. Marbouti, H. A. Diefes-Dux, K. Madhavan, "Models for early prediction of at-risk students in a course using standards-based grading," *Computers & Education*, vol. 103, pp. 1-15, 2016. https://doi.org/10.1016/j.compedu.2016.09.005.

[15] L. C. Liñán, Á. A. J. Pérez, "Educational data mining and learning analytics: differences, similarities, and time evolution," *International Journal of Educational Technology in Higher Education*, vol. 12, no. 3, pp. 98-112, 2015. https://doi.org/10.7238/rusc.v12i3.2515.

[16] Z. Kovacic, "Early prediction of student success: Mining students' enrolment data," *Proceedings of the Informing Science and IT Education Conference (InSITE)*, pp. 647–65, 2010. https://doi.org/10.28945/1281.

[17] C.-F. Tsai, C.-T. Tsai, C.-S. Hung, and P.-S. Hwang, "Data mining techniques for identifying students at risk of failing a computer proficiency test required for graduation," *Australasian Journal of Educational Technology*, vol. 27, no. 3, 2011. https://doi.org/10.14742/ajet.956.

[18] S. Ismail, S. Abdullah, "Design and implementation of an intelligent system to predict the student graduation AGPA," *Australian Educational Computing*, vol. 30, no. 2, 2015.

[19] I. E. Livieris, K. Drakopoulou, V. T. Tampakas, T. A. Mikropoulos, P. Pintelas, "Predicting secondary school students' performance utilizing a semi-supervised learning approach," *Journal of Educational Computing Research*, vol. 57, no. 2, pp. 448-470, 2019. https://doi.org/10.1177/0735633117752614

[20] P. Viswanath, T. H. Sarma, "An improvement to k-nearest neighbor classifier," *Proceedings of the 2011 IEEE Recent Advances in Intelligent Computational Systems*, 2011, pp. 227-231. https://doi.org/10.1109/RAICS.2011.6069307.

[21] X. Yang, L. Tan, L. He, "A robust least squares support vector machine for regression and classification with noise," *Neurocomputing*, vol. 140, pp. 41-52, 2014. https://doi.org/10.1016/j.neucom.2014.03.037.

[22] Z. Yu, L. Li, J. Liu, G. Han, "Hybrid adaptive classifier ensemble," IEEE Trans. Cybern., vol. 45, no. 2, pp. 177–190, 2014. https://doi.org/10.1109/TCYB.2014.2322195.

[23] Priyanka and D. Kumar, "Decision tree classifier: A detailed survey," *Int. J. Inf. Decis. Sci.*, vol. 12, no. 3, pp. 246–269, 2020. https://doi.org/10.1504/IJIDS.2020.108141.

[24] I. Taheri, M. Mammadov, "Learning the naive Bayes classifier with optimization models," *Int. J. Appl. Math. Comput. Sci.*, vol. 23, no. 4, pp. 787–795, 2013. https://doi.org/10.2478/amcs-2013-0059.

[25] J. V Shi, W. Yin, S. J. Osher, "Linearized Bregman for l1-regularized logistic regression," *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 2013, p. 1-3.

[26] C. Zhang, Y. Ma, *Ensemble Machine Learning: Methods and Applications*, Springer, 2012. https://doi.org/10.1007/978-1-4419-9326-7.

[27] T. Chen, C. Guestrin, "Xgboost: A scalable tree boosting system," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785-794, 2016. https://doi.org/10.1145/2939672.2939785.

[28] K. Fawagreh, M. M. Gaber, E. Elyan, "Random forests: from early developments to recent advancements," *Syst. Sci. Control Eng. An Open Access J.*, vol. 2, no. 1, pp. 602–609, 2014. https://doi.org/10.1080/21642583.2014.956265.

[29] S. A. Abdullah, A. Al-Ashoor, "An artificial deep neural network for the binary classification of network traffic," *Int. J. Adv. Computer. Sci. Appl.*, vol. 11, no. 1, pp. 402-408, 2020. https://doi.org/10.14569/IJACSA.2020.0110150.

[30] M. S. Alam, S. T. Vuong, "Random forest classification for detecting android malware," *Proceedings of the 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, 2013, pp. 663–669. https://doi.org/10.1109/GreenCom-iThings-CPSCom.2013.122.

[31] G. Sahoo, Y. Kumar, "Analysis of parametric & non parametric classifiers for classification technique using WEKA," *International Journal of Information Technology and Computer Science (IJITCS)*, vol. 4, no. 7, p. 43, 2012. https://doi.org/10.5815/ijitcs.2012.07.06.

[32] K. Jiang, J. Lu, K. Xia, "A novel algorithm for imbalance data classification based on genetic algorithm improved SMOTE," *Arab. J. Sci. Eng.*, vol. 41, no. 8, pp. 3255–3266, 2016. https://doi.org/10.1007/s13369-016-2179-2.

[33] S. Singh, P. Gupta, "Comparative study ID3, cart and C4. 5 decision tree algorithm: a survey," *International Journal of Advanced*

Information Science and Technology (IJAIST), vol. 27, no. 27, pp. 97-103, 2014.

[34] H.-W. Liao, D.-L. Zhou, "Review of AdaBoost and its improvement," *Jisuanji Xitong Yingyong – Computer Systems and Applications*, vol. 21, no. 5, pp. 240-244, 2012.

***AHMED AL-ASHOOR** received his BSc degree in Computer Science from Shatt-Al-Arab University College in 2010. He received his MSc degrees in computer science from Tambov State Technical University (TSTU) in 2017. Currently, he is working at University of Basra, Iraq, Basra. His research interests include data mining, network security, IoT, e-learning and fuzzy inference systems.*

***SHUBAIR ABDULLA** received his BSc degree in Computer Science from Basra University in 1994. He received his MSc and PhD degrees in computer science from University Sains Malaysia (USM) in 2007 and 2014 respectively. Currently, he is working at Sultan Qaboos University, Oman, Muscat. His research interests include data mining, network security, and fuzzy inference systems.*

•••