Predicting Academic Performance using a Multiclassification Model: Case Study

Alfredo Daza Vergaray[1], Carlos Guerra[2], Noemi Cervera[3], Erwin Burgos[4]
Professor, School of Systems and Computer Engineering, Universidad Nacional del Santa, Ancash, Perú[1, 2]
School of Systems and Computer Engineering, Universidad Nacional del Santa, Ancash, Perú[3, 4]

***Abstract*—Now-a-days predicting the academic performance of students is increasingly possible, thanks to the constant use of computer systems that store a large amount of student information. Machine learning uses this information to achieve big goals, such as predicting whether or not a student will pass a course. The main purpose of the work was to make a multiclassifier model that exceeds the results obtained from the machine learning models used independently. For the development of our proposed predictive model, the methodology was used, which consists of several phases. For the first step, 557 records with 25 characteristics related to academic performance were selected, then the preprocessing was applied to said data set, eliminating the attributes with the lowest correlation and those records with inconsistencies, leaving 500 records and 9 attributes. For the transformation, it was necessary to convert categorical to numerical data of four attributes, being the following: SEX, ESTATUS_lab_padre, ESTATUS_lab_madre and CONDITION. Having the data set clean, we proceeded to balance the data, where 1,167 data were generated, using the 2/3 for training and the remaining 1/3 for validation, then the following techniques were applied: Extra Tree, Random Forest, Decision Tree, Ada Boost and XGBoost, each obtained an accuracy of 57.41%, 61.96%, 91.44%, 59.65% and 83.3% respectively. Then the proposed model was applied, combining the five algorithms mentioned above, which reached an accuracy of 92.86%, concluding that the proposed model provides better accuracy than when the models are used independently meaning that it was the one that obtained the best result.**

Keywords*—Learning machine; prediction; academic performance; hybrid model; classification techniques; multiclassification; python

I. Introduction

The performance of a student, over the years, has always been of great importance to the institutions that provide teaching, which is why much research is done on academic achievement.

On the latter, [1] he states that "it is of great importance to support the development of students and improve the quality of higher education, which ultimately improves the reputation of institutions" (p. 21). Therefore, education plays a very important role in the progress of any society, where learning outcomes are seen as an indicator related to better health, social and more effective careers and a factor of improvement of families and communities [2].

According to [3] they indicate that about 25% of every 100 students at the higher level abandon their training in the first semester. Most of them start with failed subjects and low averages, in the third semester there is a dropout rate of 36%, a figure that increases semester by semester, until reaching 46%, which makes academic performance very transcendental and important. These results show that today's young people have the minimum of the skills needed to perform capable in contemporary societies; they have serious deficiencies to start their professional studies and of course they will have serious problems to successfully insert themselves both into the labor market and into the social, scientific, political and business groups that run the country.

Likewise, universities undertake to update their study plans and programs to adapt them to the needs of today's society; Unfortunately, while these efforts are important, modifying or changing the curriculum does not eliminate learning problems, but also presents new challenges. Similarly, according to [4] to consider a university as one of high quality it is necessary that it has an excellent record of academic achievement.

As an idea of solution to achieve this goal, is that the use of new technologies is becoming more and more frequent, as is the case of data mining.

According to [5] "Data mining is a process of automatically extracting useful information from large data set repositories, etc." In addition, the usefulness of this technology is that it can be used to train learning models, which from historical data can discover useful learning information and based on this, make a prediction [6].

Currently this technology is applied in various fields such as industry, banking, among others. Applied to the field of education, it is called Educational Data Mining (EDM), which, according to [7] "is an emerging area of research composed of a large set of psychological and computational approaches to provide a roadmap of how students learn." On the other hand, at present the existence and constant use of automated learning tools allow, according to [7] to store "a variety of data related to students and valuable characteristics that affect the performance of students and that can be used in the construction of the prediction model".

In Peru, the problem of low performance is frequent at different levels of education. The performance of the students of the different engineering faculties of the Universidad Nacional del Santa is medium low of the vast majority according to the report made by the [8], this affects both the university, since "its success depends on the success of its students" according to [9], and the students themselves, since

they opt for desertion, career change, or limit them when it comes to finding work in the future, where in their studio they developed a predictive model using the J48 technique, which obtained an accuracy of 60.9%.

Currently there are many learning models that are used to extract knowledge from a data set, some of these are those based on Naive Bayes (NB), Vector Support Machine (SVM), Decision Trees (DT), the Closest Neighbor (KNN), among others; but as stated [10], it is difficult to find an efficient classification model that can be used for various situations or problems. That is why the idea of combining several classifiers (multiclassifiers) was born.

The multiclassifiers according to [18] "belong to a recent area of data mining that has allowed to improve, in general, the accuracy of predictions through the combination of individual classifiers" and some of these multiclassifiers are Streaming Ensemble Algorithm (SEA), Coverage Based Ensemble Algorithm (CBEA), multiclassification based on CIDIM (MultiCIDIM-DS) and MultiCIDIM-DS-CFC.

Based on the problem that arises in the university and in search of improving the various solutions proposed by several studies, it was proposed to create a hybrid model that is capable of predicting academic performance so that students and teachers can opt for preventive measures to avoid that grades are deficient in the future.

This article aims to create a predictive model making use of multiclassification through the Stacking technique using new algorithms such as: Extra Tree, Random Forest, Decision Tree,Ada Boost and XGBoost to achieve better accuracy, taking into account that the studies that have been done so far, make use of a single prediction technique, thus generating a good precision, but that could be better if several techniques were applied together.

The rest of the work is structured in five sections. In Section II, a review of the literature of related works is presented. Section III contains the method, which outline the data mining process implemented in this study, which includes a representation of the collected dataset, an exploration and visualization of the data, and finally the implementation of the data mining tasks and the final results. Section IV shows the findings and discussion obtained after the creation and testing of the predictive model. Finally, Section V contains the conclusions reached after the development of the model.

II. Related Work

This section compiles various research conducted in recent years on the application of data mining in the education sector.

There are many studies or works carried out related to education, whose main theme is the academic performance or dropout of students such as the study carried out by [8] whose objective was to compare various data mining techniques when using them to predict the performance of students. The data they used was from the Kaggle repository and this comparison included the techniques: Decision Tree (C5.0), Naïve Bayes, Random Forest, Support Vector Machine, K-Nearest Neighbor and Deep Neural Network, this being the one that obtained a greater precision, reaching 84% accuracy.

Another work is also carried out by [11], aimed to perform a comparison and study of hybrid classification model and machine learning algorithms based on decision tree, clustering, artificial neural network, Naïve Bayes, etc., using the open source data mining tool Weka for a practical experiment on a student dataset, having as results that the hybrid method achieved the highest accuracy of 92.59% than individual classifiers, that is, J48, NB, IBK and ANN achieved an accuracy of 85.18, 81.48, 88.88 and 88.88%.

In [12], the authors developed a regression model to predict the score that a student would have, used the ALGORITHM KNN, Decision Tree, SVM, Random Forest and Multiple Linear Regression. After comparing the results of each algorithm, it was the Multiple Linear Regression Model that obtained the greatest accuracy. In [13], They conducted a study of student dropout to determine what were the causative factors and which classification algorithm is the most used to predict this problem. After reviewing several studies, they concluded that decision tree classifiers were the most commonly used, as they obtained good predictions.

A semi-supervised learning approach is the one they used [14] to rank the performance of first-year college students. The categories to classify were low, medium and high and the classifier was Naive Bayes, who obtained an accuracy of 96% and specificity of 100%.

In [15], they build a model that predicts the outcomes students will achieve in the semester. They used 13 learning algorithms, belonging to 5 categories, for the Bayes category, they used Naive Bayes, for the Function category they used SVM and Perceptron Multilayer, for the Lazy category, the IBK technique, for the Rules category they used Decision Table, JRip, OneR, Part and ZeroR and finally for the Trees category, they used the J48 techniques, Random Forest, Random Tree, and Simple CART. The data correspond to 50 students and they developed their model on the Weka platform, after the results of having applied each of these techniques, the one that had the best results was the J48 technique, which reached 88% accuracy in the prediction

In the same way the study carried out by [16], whose objective is to develop a prediction model based on Bayes, specifically Naive Bayes and Bayes Network. The data was collected through a questionnaire of 62 questions related to health, social activity, relationships and academic performance. They used the Weka tool in which they obtained as a result the algorithm Naive Bayes is better, since it obtained 70.6%, while Bayes Network obtained 64.3% accuracy.

In [17], the author in his research aimed to develop an algorithm with incremental learning to mine data flows that is capable of manipulating gradual, abrupt or recurrent concept changes, obtaining as results that the FAE algorithm achieved promising results in the tests, compared to well-known algorithms implemented also in the MOA work environment, taking into account the parameters: Accuracy (82.4%), execution time, behavior in the transition period from one concept to another and recovery time after a change of concept.

In the present study, five classification algorithms are used as the basis for the creation of a multiclassifier model, through

the Stacking technique, these algorithms being: Extra trees, Random Forest, Decision Tree, Ada Boost, XGBoost.

A. Extra Trees

It is the short name of Extremely Randomized Trees, which means Extremely Randomized Trees. This technique consists of a large number of individual decision trees. It is characterized because it uses the entire set of training data, to grow each decision tree [18].

The Extra Trees algorithm creates many decision trees at random, with the intention of finding a final answer, from the combination of the results of each tree. The difference with the Random Forest algorithm, which has the same procedure, is that the number of random processes used in Extra Trees is much higher.

B. Random Forest

A random forest consists of many decision trees. Each tree in the forest is a binary tree and its generation follows the principle of top-down recursive division [19]. For each tree, the root node contains all the training data and this is divided into two nodes, the left and right, according to certain rules and these in turn train with different samples of data. The division continues to occur based on certain rules until the fork stop is met.

C. Decision Tree

The decision tree is a tree-like structure that represents a series of decisions and the resulting decision takes the form of rules for classifying a given data set [20]; these are supervised algorithms that can be used for both classification and regression. The objective of this algorithm is to predict by learning decision rules. After the construction of a decision tree, these classify an instance from the root node of the tree then it is directed to a leaf of the tree that would be the intermediate node, depending on the value it takes and this is done successively until it reaches the last leaf of the tree that would be the terminal node.

D. Ada Boost

The Ada Boost algorithm stands for Adaptive Boosting and is one of the most popular techniques of the Boosting method. This algorithm is iterative and its operation consists of training different classifiers considered weak for the same set of training data, then combining them to form a stronger classifier [21].

Ada Boost classifiers represent a robust class of classifiers that aim to increase or improve the accuracy of an already built classifier [14].

E. XGBoost

XGBoost is an advanced software based on Gradient Tree Boosting that can efficiently handle large-scale machine learning tasks [22]. The XGBoost algorithm stands for Extreme Gradient Boosting and is a supervised algorithm based on the Boosting method. To achieve the strongest classifier, an optimization algorithm is used, Gradient Descent and each model generated is compared with the previous one and if a new model obtains better accuracies, this is taken as a basis for relicensing modifications. But, if in case, its accuracies are low, it returns to the previous model, making modifications based on this. The process is repeated until the differences between two consecutive models are negligible, which means that the maximum number of iterations was reached.

F. Stacking

Set learning algorithms are metaalgorithms that combine different machine learning algorithms into a single predictive model to reduce bias (boosting), variance (bagging) or improve the accuracy of predictions (stacking) [23]. Based on the study presented in this section a stacking method was developed (see Fig. 1). This technique involves using predictions from previous-level machine learning models as input variables for next-level models [5].

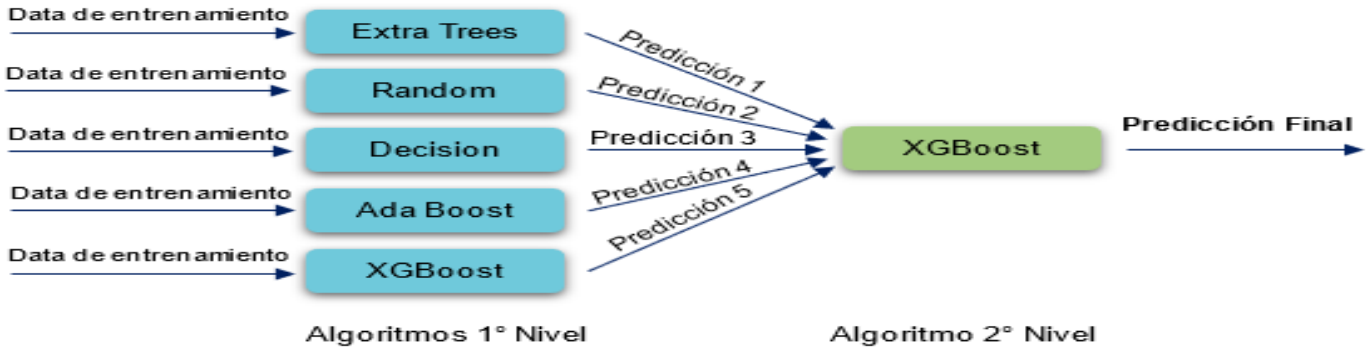

Fig. 1. Graphical Representation of the Stacking Method used in this Work.

III. METHOD

For the development of the present study, the steps of the stacking method shown in Fig. 2 have been followed, it should be noted that XGBoost is an advanced software based on Gradient Tree Boosting that can efficiently handle large-scale machine learning tasks [14].

1) Data integration: The data of systems and computer engineering and agroindustrial engineering were integrated.

2) Selection: This work focuses on students of systems engineering, energy and agro-industrial of the Universidad Nacional del Santa. Data were collected using an online questionnaire, which included questions related to some characteristics about academic performance. The questionnaire contained a total of 25 characteristics and were answered by 557 students, of whom 135 were female and 422 were male. The characteristics considered in the questionnaire are detailed in Table I.

3) Pre-processing: This step is important to prepare the data before it is used in testing. In our case, the data required pre-processing, as there was empty data, inaccurate data and irregular or inconsistent data. Some of the tasks included in this step are: data cleansing, transformation data, reduction data, and integration data [19]. Another detail of the collected data set is that they are mostly categorical, and for this data to be used in the selected tool, Python, it must necessarily be numerical data.

a) Removing attributes: Initially, the characteristics SCHOOL, Cod_student, CI_ante, prom_trans, NATIONALITY, FECH_nac, RACE, TYPE_viv, PLACE_res and other attributes that do not necessarily have a great correlation with the student's performance were eliminated, as shown in Table II.

b) Data cleansing: Data cleansing required deleting records that contained empty or inconsistent data. In the first instance, there were 88 records that had at least one empty value, after their elimination, there was a record that contained an invalid data, finally leaving 500 records to be used in the model to be proposed.

c) Creating the output class: The focus of this report is classification, and taking into account that the collected data set had an attribute, AVERAGE_ACU, which contained the academic averages of the students surveyed, the creation of 3 categories was considered so that the model can classify a certain student in one of those categories. These three categories were considered based on the following:

- Bad, whose rating is less than 10.5.
- Regular, whose rating is greater than or equal to 10.5, but less than 14.
- Regular, whose rating is greater than or equal to 10.5, but less than 14.
- Well, whose rating is greater than or equal to 14, but less than or equal to 20, this value being the maximum in Peru's rating system (vigesimal system).

Therefore, the final output class considered for the classification model is, CONDITION, the following attributes as shown in Table III.

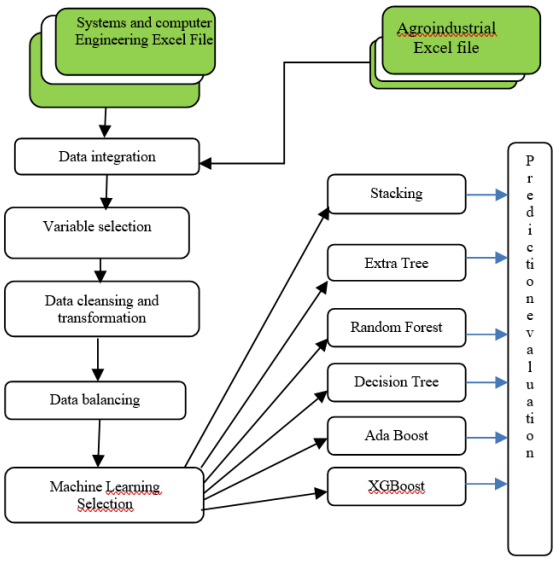

Fig. 2. Process of Prediction of Academic Performance through the Stacking Model.

TABLE I. LIST OF FEATURES USED IN THE QUESTIONNARE

Feature	Description	Feature	Description
SCHOOL	Academic school	ESTATUS_lab_padre	Employment status of the father
Cod_student	Student Code	ESTATUS_lab_madre	Mother's employment status
SEX	Gender	TYPE_viv	Type of housing
CI_ante	Previous cycle	INCOME_pa	Father's income
AVERAGE_acu_	Academic performance	INCOME_ma	Income of the mother
Prom_trans	Previous average	PLACE_res	Place of residence
NATIONALITY	Nationality	scholarship	Do you have a scholarship?
CURRENT_AGE	Current age	c_otra_carr	Do you have another career?
Anio_ingreso	Year of admission to the University	c_title_otra	Do you have another title?
AGE_estudiar	Age at which he began to study	DEPARTMENT	Department
FECH_nac	Date of birth	PROVINCE	Province
RACE	Race	DISTRICT	District
n_int_fami	Number of members in the household		

TABLE II. LIST OF FEATURES WITH THE HIGHEST CORRELATION

Feature	Domain
SEX	*Nominal (Female, Male)*
CURRENT_AGE	*Whole*
Anio_ingreso	*Whole*
AGE_estudiar	*Whole*
N_int_fami	*Whole*
ESTATUS_lab_padre	*Nominal (Dependent, Independent)*
ESTATUS_lab_madre	*Nominal (Dependent, Independent)*
INCOME_pa	*Real*
INCOME_ma	*Real*
AVERAGE_acu	*Real*

TABLE III. LIST OF FINAL FEATURE

Feature	Domain
SEX	Nominal (Female, Male)
CURRENT_AGE	Whole
Anio_ingreso	Whole
AGE_estudiar	Whole
N_int_fami	Whole
ESTATUS_lab_padre	Nominal (Dependent, Independent)
ESTATUS_lab_madre	Nominal (Dependent, Independent)
INCOME_pa	Real
INCOME_ma	Real
CONDITION	Nominal (Bad, Regular, Good)

d) Data balancing: Fig. 3 shows that the dataset is unbalanced, so accuracy could be affected. There is a lot of difference between the minority class (Good) and the majority class (Regular), so it was necessary to balance the dataset to ensure a better percentage of accuracy. The technique of oversampling has been used for data balancing as shown in Fig. 4, which generates artificial examples of the minority class, until reaching the number of records of the majority class.

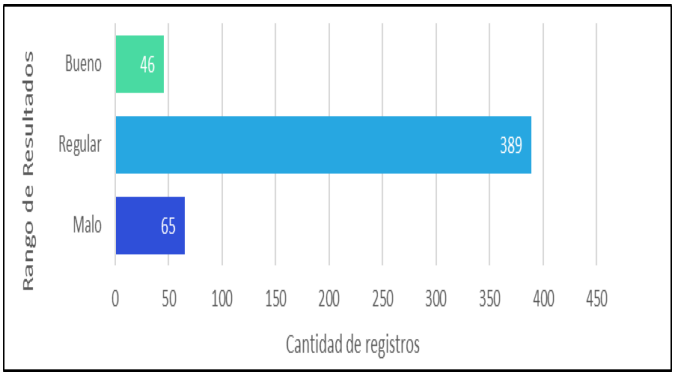

Fig. 3. Output Class Records before Applying Data Balancing.

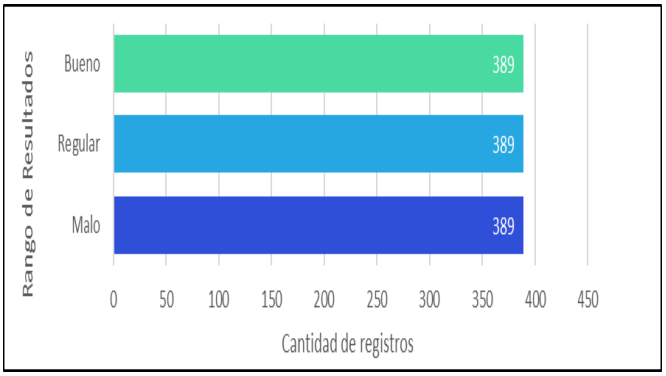

Fig. 4. Output Class Records after Applying Data Balancing.

4) Transformation: All development and implementation of the predictive model and data processing was done using Python software.

Due to the use of this tool, it was necessary for the entire dataset to have numeric values. That is why the data of the characteristics SEX, ESTATUS_lab_father, ESTATUS_lab_ mother and CONDITION, had to be transformed into numbers according to conditions. Table IV shows the new values of the characteristics that were transformed.

5) Data mining: Python is a tool for data mining that offers many modules or libraries to be used professionally. These modules help the application of various classifiers.

For the application of data mining, one of the ways to increase the accuracy of the prediction made by a specific classification technique is the use of ensemble learning algorithms, one of them being stacking, which is what was applied for the construction of this predictive model.

According to the definition of [24] "it's the process of using different machine learning models one after another, where the predictions of each model are aggregated to create a new feature."

The techniques used as a basis for using stacking were: Extra trees, Random Forest, Decision Tree, Ada Boost, XGBoost.

TABLE IV. LIST OF FEATURES WITH NUMERIC VALUES

Feature	Numeric values
SEX	1 = Female 2 = Male
ESTATUS_lab_father	1 = Dependent 2 = Independent 3 = Deceased 4 = Live without a father
ESTATUS_lab_mother	1 = Dependent 2 = Independent 3 = Housewife 4 = Not working
CONDITION	1 = Bad 2 = Regular 3 = Good

The dataset was classified into two groups. The first group of data is for training and is made up of 2/3 of the total data. The second group is the test group and is made up of the remaining 1/3 of data. The application of the stacking algorithm was done using the training data to prepare the model, and then the test data. Model performance can be observed after application of the model to the test dataset.

6) Interpretation and evaluation: Python is used to import the dataset from an excel spreadsheet. The attributes with the highest correlation were selected to apply model training. The attributes sex, father's employment status, and mother's employment status were converted to numerical to avoid errors during the modeling process. Applied the different techniques to the final data set, each of these obtained different precisions.

To know the efficiency of the classifiers, these were evaluated using a confusion matrix, in which the number of records classified correctly and incorrectly is appreciated. 5 models were built to individually analyze the performance of the models and each of these obtained the following matrix confusion:

a) Extra trees: The first model to be built was the model based on the Extra Trees sorter. For its application in the Python tool, the ExtraTreesClassifier class of the sklearn.ensemble library was used and the following parameters were considered:

- random_state = 0
- n_jobs = -1
- n_stimators = 100
- max_depth = 3

Applied the algorithm to the dataset, the confusion matrix obtained was as follows:

As shown in Table V, is classified most students in the Bad category. These results show that the algorithm does not give good accuracy for this case.

b) Random Forest: For the implementation of this model, the RandomForestClassifier class of the sklearn.ensemble library was used and the parameters considered for its application were the following:

- random_state = 0
- n_jobs = -1
- n_stimators = 100
- max_depth = 3

TABLE V. CONFUSION MATRIX FOR THE EXTRA TREES MODEL

		Prediction		
		Bad	Regular	Well
Current	Bad	108	4	2
	Regular	96	16	22
	Well	97	2	42

The confusion matrix that was obtained from this model was as follows:

Based on the data obtained in Table VI, it can be said that the model has predicted well in terms of the classification of students in the Bad and Good categories, but for the Regular category, there is some uncertainty when classifying almost the same number of students for the three categories.

c) Decision Tree: To implement this algorithm, the DecisionTreeClassifier class of the sklearn.tree library was used and the parameters considered were the following:

- random_state = 0
- min_samples_split = 2
- max_depth = None

Running this model gets the following confusion matrix:

According to the values of the matrix, this is the model that has obtained a better precision, because as shown in Table VII, of 114 students correctly predicted the 114 within the Bad category, of 134 students correctly predicted 96 within the Regular category, and of the rest, 15 incorrectly predicted as Bad and 23 as Good. Finally, he correctly predicted 141 students as Good.

d) Ada Boost: This algorithm was implemented using the AdaBoostClassifier class of the sklearn.ensemble library and the parameters considered for its application were the following:

- random_state = 0
- n_stimators = 100

The confusion matrix that was obtained from this model was as follows:

Table VIII shown that of each dataset belonging to the three categories, the model was able to classify half of the students correctly, but not enough to be considered a good prediction, since it misclassified a large percentage of the students. So it is assumed that accuracy is not good for this model.

TABLE VI. CONFUSION MATRIX FOR THE RANDOM FOREST MODEL

		Prediction		
		Bad	Regular	Well
Current	Bad	87	9	18
	Regular	42	51	41
	Well	34	12	95

TABLE VII. CONFUSION MATRIX FOR THE DECISION TREE

		Prediction		
		Bad	Regular	Well
Current	Bad	114	0	0
	Regular	15	96	23
	Well	0	0	141

TABLE VIII. CONFUSION MATRIX FOR THE ADA BOOST MODEL

		Prediction		
		Bad	Regular	Well
Current	Bad	62	43	9
	Regular	27	77	30
	Well	9	45	87

e) XGBoost: For the implementation of this model, the XGBClassifier class of the xgboost library was used and the parameters considered for its application were the following:

- random_state = 0
- n_jobs = -1
- learning_rate = 0.1
- n_stimators = 100
- max_depth = 3

The confusion matrix obtained from this model was as follows:

This algorithm is the second most accurately after the decision tree. Table IX shown that of 114 students who belonged to the Bad category, it correctly ranked 108. Out of 141 students in the Good category, he correctly predicted 132. But of the 134 students considered regular, only 80 could predict correctly.

TABLE IX. CONFUSION MATRIX FOR THE XGBOOST MODEL

		Prediction		
		Bad	Regular	Well
Current	Bad	108	1	5
	Regular	23	80	31
	Well	0	9	132

What is part of a confusion matrix are the following four classifiers:

- True Positives (TP): These are the records correctly classified in the positive class.
- False Positives (FP): These are the records incorrectly classified in the positive class.
- False Negatives (FN): These are the records incorrectly classified in the negative class.
- True Negatives (TN): These are the records classified correctly in the negative class.

From these values, the following metrics can be calculated to evaluate the effectiveness of a predictive model:

- Sensitivity

This metric measures the positive values, in this case, correctly identifying students in the Bad, Regular and Good categories, according to the given parameters.

$$Sensitivity(TPR) = \frac{TP}{TP + FN}$$

- Specificity

This metric measures the negative or false values.

$$Specificity\ (TNR) = \frac{TN}{TN + FP}$$

- Precision

This metric measures the total number of items correctly classified as positive.

$$Precision\ (P) = \frac{TP}{TP + FP}$$

- Accuracy

$$Acuracy(ACC) = \frac{TP + TN}{TP + FP + FN + TN}$$

This metric measures the veracity of the prediction, that is, the difference between the predicted value and the actual one.

According to the results obtained from the application of the techniques individually, each of these obtained the following percentages in their precisions, shown in Table X:

TABLE X. METRIC RESULTS FOR INDIVIDUAL TECHNIQUES

Classification Technique	**Metric**			
	Sensitivity	**Specificity**	**Precision**	**Accuracy**
Extra trees	.4548	.7295	.5741	.6178
Random Forest	.6058	.8011	.6196	.7326
Decision Tree	.9054	.9509	.9144	.9348
Ada Boost	.5785	.7889	.5965	.7206
XGBoost	.8268	.9106	.8330	.8817

And after the application of stacking, the results he obtained were shown below in Table XI:

TABLE XI. METRICS RESULTS FOR STACKING

Classification Technique	**Metric**			
	Sensitivity	**Especificidad**	**Sensitivity**	**Exactitud**
Stacking	.9253	.9619	.9286	.9485

7) Knowledge: A graphical interface was created (Fig. 5) in which the previously developed predictive model was integrated, for the ease of use of teachers and in this way the performance of new engineering students can be predicted.

As a test, the following values were entered into the interface:

Age:

Sex:

Members:

And the result obtained concludes that the student will obtain a low academic performance.

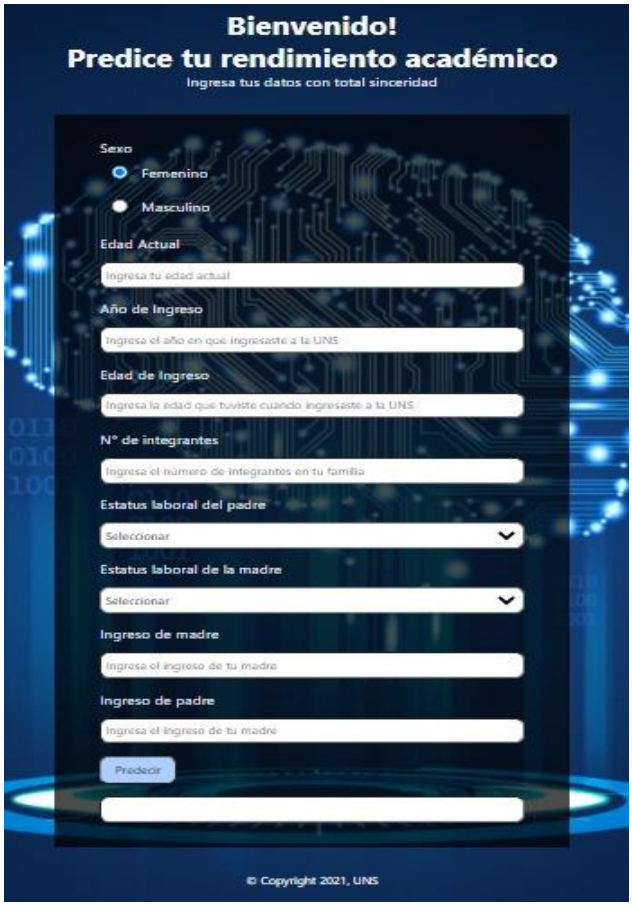

Fig. 5. Web Interface.

IV. Findings and Discussion

In this paper, the stacking technique was used to predict students' academic performance. In addition, as part of this technique, other classification methods such as Extra Trees, Random Forest, Decision Tree, AdaBoost and XGBoost were necessary, which are algorithms that obtain good results for a certain type of situation. During the modeling process, a cross-validation of 10 was required to ensure that each result is independent of division for training and test data.

The results obtained from the stacking technique (which is a combination of five algorithms) is 92.86% accuracy which is very encouraging with respect to the other results obtained by Extra Tree with 57.41% accuracy, Random Forest 61.96%, Decision Tree 91.44%, Ada Boost 59.65% and XGBoost with an accuracy of 83.3%, while authors such as [15], in their study show that the Naives bayes algorithm gave as a significant useful result an accuracy of 84%, being considered the best algorithm to predict academic performance and thus be able to arrive at solutions to improve the problem. On the other hand, in the study of the researchers [9], they developed a predictive model using the Rep Tree technique, which obtained an accuracy of 60.9%.

With regard to sensitivity, it is so that in the present study the Extra Trees technique was used, which reached a sensitivity of 45.48%, the Random Forest technique, 60.58%, the Decision Tree technique 90.54%, Ada Boost 57.85% and finally XGBoost obtained a sensitivity of 82.68%, however when combining the aforementioned techniques through the Stacking technique a sensitivity of 92.53% was obtained, exceeding the percentage of sensitivity of the models; this is corroborated with the study of [16], which using the Naive Bayes technique, the model obtained a sensitivity of 66.7%.; they also [6] did a study in which a sensitivity of 88.7% was reached using Random Forest.

Likewise, with regard to specificity using the extra trees, Random Forest, Decision Tree, Ada Boost, XGBoost and each of these techniques, a specificity of 72.59%, 80.11%, 95.09%, 78.89%, 91.06% respectively was obtained and when performing the combination through the Stacking technique a specificity of 96.19% was obtained. In a study conducted by [14] they made use of the Naive Bayes technique, which obtained a specificity of 100%.

A limitation of the present study is that it was carried out considering the total average of the academic cycle of the students and in addition, they only belonged to the engineering schools. Results may vary for other schools or if a specific course or subject is considered.

Universities can employ the generation of a predictive model through stacking to predict the results of students' academic performance by cycle. Since it was demonstrated that its use and application can achieve a better accuracy in the prediction. This will help improve academic performance, as it will allow corrective action to be taken in advance and will also help reduce the percentage of students suffering from an academic delay.

V. Conclusion

The objective of this work is to develop a model that achieves better accuracy compared to the individual application of various techniques, through the stacking of different classification techniques, and in this way check if a better prediction is obtained.

The questionnaire applied to students to obtain the data set contains many questions, some of which have a greater correlation with academic performance than others. So maintaining the most important features benefits the accuracy of the prediction.

To achieve a good percentage of accuracy, it was necessary to have a process that involved cleaning the data, eliminating attributes less correlated with the output class, eliminating incomplete records or with invalid values. In this model, 5 classification techniques are used that are part of the multiclassification technique, stacking. The individual techniques were Extra Trees, Random Forest, Decision Tree, AdaBoost and XGBoost, after obtaining the predictions of each technique, XGBoost was used as a second-level technique to make the final prediction.

After having applied stacking it can be concluded that this has given better results than the application of the techniques individually.

References

[1] D. Ha, C. Giap and N. Huong, “An empirical study for student academic performance prediction using machine learning techniques”,

International Journal of Computer Science and Information Security.Vietnam, vol.18, no. 3, pp. 21-28, March 2020.

[2] D. Rodríguez and R. Guzmán, "Rendimiento académico y factores sociofamiliares de riesgo. Variables personales que moderan su influencia", Perfiles educativos. Perú, vol.41, no.164, pp.118-134 , Abril 2019.

[3] M. Flores, H. Rivera and F. Sánchez, "Bajo rendimiento académico : más allá de los factores sociopsicopedagógicos", *Revista Digital Internacional de Psicología y Ciencia Social. México,* vol. 2, no.1, pp. 95-104, Enero 2016.

[4] S. Vadivukkarasi and S. Santhi. "A novel hybrid learning based Ada Boost (HLBAB) classifier for channel state estimation in cognitive networks". International Journal of Dynamics and Control. India, pp. , 299-307, April 2021.

[5] B. Pavlyshenko, "Using Stacking Approaches for Machine Learning Models", 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP).Ukraine, pp. 25-28, August 2018.

[6] D. Aggarwal, S. Mittal, and V. Bali, "Prediction Model for Classifying Students Based on Performance using Machine Learning Techniques", International Journal of Recent Technology and Engineering (IJRTE). Indian, vol. 8, no. 257, pp. 496-503, July 2019.

[7] S. Bhutto, I. Farah, Q. Ali and M. Anwar, "Predicting Students' Academic Performance Through Supervised Machine Learning" , 2020 International Conference on Information Science and Communication Technology (ICISCT). Pakistan, pp. 1-6, February 2020.

[8] S. Nageswari, M. Pallavi and P. Divya, "Comparison of classification techniques on data mining", International Journal of Emerging Technology and Innovative Engineering.India, vol.5, no.5, pp. 267-272, April 2019.

[9] A. Hamoud, A. Hashim and W. Awadh, "Predicting student performance in higher education institutions using decision tree analysis", International Journal of Interactive Multimedia and Artificial Intelligence. Iraq , vol. 5,no.2 , pp. 26-31, February 2018.

[10] OEI. "Indicadores Socio-Económicos de los estudiantes de pregrado de la UNS", Nuevo Chimbote: Universidad Nacional del Santa, 2019.

[11] K. Rawat and I. Malhan, "A Hybrid Classification Method Based on Machine Learning Classifiers to Predict Performance in Educational Data Mining", Proceedings of 2nd International Conference on Communication, Computing and Networking. Switzerland, vol.46, no.1., pp. 277-68, September 2019.

[12] N. Chauhan, K. Shah, D. Karn and J. Dalal, "Prediction of Student's Performance Using Machine Learning", 2nd International Conference on Advances in Science & Technology (ICAST). India, pp.1-5, April 2019.

[13] S. Ahmad, S. Mutalib, H. Abdul and S. Abdul, "A Review on Student Attrition in Higher Education Using Big Data Analytics and Data Mining Techniques", I.J. Modern Education and Computer Science. Malaysia, vol. 11, no. 8, pp. 1-14, August 2019.

[14] Y. Widyaningsih, N. Fitriani and D. Sarwinda, "A Semi-Supervised Learning Approach for Predicting Student's Performance: First-Year Students Case Study", 2019 12th International Conference on Information & Communication Technology and System (ICTS). Indonesia, pp. 291-295. July 2019.

[15] I. Khan, A. Al Sadiri, A. Ahmad and N. Jabeur, "Tracking Student Performance in Introductory Programming by Means of Machine Learning", 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC). Malaysia,pp.1-6, January 2019.

[16] A. Hamoud, A. Humandi, A. Awadh, and A. Hashim, "Students' success prediction based on Bayes algorithms", International Journal of Computer Applications. Iraq, vol.178, no.7, pp.6-12, November 2017.

[17] A. Ortiz, "Algoritmo multiclasificador con aprendizaje incremental que manipula cambios de conceptos", Granada: Universidad de Granada, 2014.

[18] M. Camana, S. Ahmed, C. Garcia and I. Koo, "Extremely Randomized Trees-Based Scheme for Stealthy Cyber-Attack Detection in Smart Grid Networks", IEEE Access. Korea, vol. 8, no.1 ,pp. 19921-19933, January 2020.

[19] Y. Xiang, L. Li and W. Zhou, "Random Forest Classifier for Hardware Trojan Detection", 12th International Symposium on Computational Intelligence and Design. China, pp. 134-137, December 2019.

[20] E. Irfiani, I. Elyana, F. Indriyani, F. Schaduw and D. Harmoko, "Predicting Grade Promotion Using Decision Tree and Naïve Bayes Classification Algorithms", 2018 Third International Conference on Informatics and Computing (ICIC). Indonesia, pp.1-4, October 2018.

[21] Z.Yong, L. Jianyang, L. Hui and G. Xuehui, "Fatigue Driving Detection with Modified Ada-Boost and Fuzzy Algorithm", 2018 Chinese Control And Decision Conference (CCDC).China,pp. 5971-5974, June 2018.

[22] C.Wang, C. Denga and S. Wang, "Imbalance-XGBoost: leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost", Pattern Recognition Letters. China, vol. 136, no.1, pp. 190-197, August 2020.

[23] S. Asante, P. Ngare, and D. Ikpe,"On Stock Market Movement Prediction Via Stacking Ensemble Learning Method", 2019 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr). Kenya, pp.1-8, May 2019.

[24] IBM. (17. Enero 2020). Stack machine learning models: Get better results. Von IBM Developer: https://developer.ibm.com/articles/stack-machine-learning-models-get-better-results/ abgerufen.