Platforms and Graphication Tools for a mobile application: Simulation of Planetary Objects' Trajectories

Analuz Silva Silverio[1,*], Mario A. De Leo Winkler[2], Antonio Alarcón Paredes[1]

[1]Universidad Autónoma de Guerrero, Unidad Académica de Ingeniería, Chilpancingo, Mexico
sisianlight@gmail.com,aalarcon@uagro.mx
[2]University of California, Physics and Astronomy Department, Riverside, USA
mario.deleo-winkler@ucr.edu

Abstract. During the development of mobile applications some of the first things that should be taken under consideration are the platform, code and development tools to be used. To do this, it is important to keep in mind compatibility of devices and operating systems, the focus audience, the project's budget, and other factors that end up being crucial for the successful completion of a project of this type. Because this particular case we want to develop applications for scientific outreach in astronomy, we investigate and present trends in developing mobile applications as well as propose a number of tools for our particular project, themed on planetary objects in a bi-dimensional simulation of positions, as well as the charted and tabulated data.

Keywords: Physics computing, Mobile computing, Native app, Application software, Open source software, Software tools, Programming environments, Computer science education, Astronomy.

1 Introduction

In mobile applications' (apps) development, one of the first things to take into account is the type of application that is being developed. For this reason, it becomes necessary to know positive and negative aspects in the trends of apps development.

Nowadays, the guidelines of app development have a wider scope than the initial ones where only two paradigms existed: native apps and web apps. Native applications are developed on the native language of an operating system (OS). A web application consists of a website specifically optimized with an interface and a set of functions to be used in mobile devices. Recently, with the arose of HTML5, the concept of hybrid applications has emerged and comes with added features and functionalities related to direct hardware control and the possibility to be implemented in different OS. Those new features have made hybrid apps become a new trend of development.

Unconventional Computing to Estimate Academic Performance in University Freshmen Students

Carolina Fócil Arias, Amadeo José Argüelles Cruz, and Itzamá López Yáñez

[1] Centro de Investigación en Computación
[2] Centro de Innovación y Desarrollo Tecnológico en Cómputo

focil.carolina@gmail.com, jamadeo@cic.ipn.mx, ilopez@ipn.mx

Abstract. Tracking academic performance of students with different levels is a topic of actual relevance. For undergraduate students, in the specific field of Mathematics, different techniques of pattern recognition have been applied to estimate their academic performance. In this study, we propose the use of a unconventional model called Gamma classifier for estimating academic performance, in freshmen student, applied to a dataset provided by the Technological University of Pereira, located in Colombia. The results revealed that Gamma classifier is a competitive algorithm, helping to determine whether a freshman student will pass Mathematics course.

Keywords: Academic Performance, Gamma Classifier, Machine Learning, Wilson Editing, Data Cleaning

1 Introduction

The little interest for the studies, the lack of abilities for developing solutions to reasoning problems, among others academic characteristics, are some factors that affect students performance in Mathematics course, which is related with student desertion. In spite of all services and programs efforts to avoid student desertion, only half of them get a bachelors degree [7].

Technological University of Pereira (located in Colombia) is an example of student desertion, it has records of roughly 52% of dropout between 1994 and 2004, where, 18.3% of students left the university due to poor academic performance in Mathematics in the first year[4].

This research is oriented as an alternative, applied with other techniques based on data and evidence, to reduce the dropout in freshmen engineering study and take actions over needed interventions in Mathematics course. We propose the use of a dataset provided by Technological University of Pereira and an unconventional algorithm called Gamma classifier which has been used in several applications [13][12]. Then, results are compared with those obtained in

[4] and other seven classifiers provided by a system called WEKA [16], with the purpose to determine whether a student will pass or no Mathematics course.

At the same time, data cleaning and Wilson editing techniques are also being applied to improve classifier performance, since dataset has many missing values.

The rest of this paper is structured as follows. Section 2 describes related work using patterns recognition approach. Section 3 describes the techniques used for determining whether a student will pass or not Mathematics course. Section 4 summarizes the results of using Gamma classifier and the techniques described in section 3. Finally, conclusions and future works are shown in section 5.

2 Related Work

Patterns recognition have been applied in several studies which have allowed to improve education in several levels and areas. In this section, we present some related studies with education and the use of several approaches to evaluate academic performance. For example, University of Singhaniya located in India [3] used a classification task to evaluate student's performance at the end of the semester in accordance with quiz, seminar, attendance and assiggnment of each student. The results obtained is based on decision tree approach.

University Technological of Pereira in Colombia [4] applied a Logistic Multiple Regresion model as proposal to identify the factors that essentially allow to predict whether a freshman engineering student will pass or not Mathematics course, in essence social and academic factors were used.

Another researches such as [7][14][2] have used Support Vector Machines into classification area. The proposed project by [7] is able to predict freshmen student attrition according to the past and present educational success. A dropout method for e-learning courses is reported by [14] which determines whether a student will pass or not Networks and Web design courses using the methods Neuronal Networks, Support Vector Machines and Probabilist approach. The School of Physical Education and Sports at Cukurova University located in Turkey [2] used Support Vector Machines to predict whether a candidate will be admitted in University of Cukurova based on a physical test.

A software called MUSKUP (Mugla University Student Knowledge Discovery Unit Program) was developed by [11] to identify essentially facts that affect the success in a student using the method of Decision Tree.

One of the most popular data mining techniques used to improve educational standards are association rules [16]. The study [1] extracted from a dataset some association rules with the aim of determining how many students are inscribed in a program but they are not interested on it. Another work [6] identified why some students did not finish their career in a period of years not less than 6 years.

3 Material and Methods

In this study, we have used an unconventional model called Gamma classifier and two preprocessing techniques: Data Cleaning and Wilson Editing with the aim of determining with higher precision whether a freshman engineering student will pass or not Mathematics course. To validate Gamma classifier performance a stratified k-fold cross validation was used. In stratified k-fold cross validation each fold contains approximately the same proportion of predictor labels as the original dataset [7].

3.1 Dataset

The dataset used in this study came from Technological University of Pereira located in Colombia with an enrollment of 834 freshmen engineering students and it was recorded during a period of 2005 to 2007. The dataset has two classes for identifying when a student pass or not Mathematics course. In this case, the class takes the value of 0 whether a student did not pass Mathematics course and it takes the value of "1" whether a student passed Mathematics course. For better interpretation, the dataset has 481 students that did not pass and 353 students that passed Mathematics course.

The variables used in this study are related with academic performance and social behavior. Table 1 shows the variables used in this experimental study and their selection is based on [4].

Table 1: Description of the variables used in this study

Variables	**Description**
ICFES	Points obtained from Colombian Institute for the Promotion of Higher Education
CodProg	Program code
Cabstract	Abstract Logical Thinking

According to Technological University of Pereira, the academic programs used in this study are Electrical Engineering, Industrial Engineering, Mechanical Engineering, Electrical Technology, Industrial Technology, Mechanical Technology, Chemical Technology, Engineering Systems and Computing, Physical engineering, Systems Engineering and Industrial Engineering.

3.2 Data Preprocessing: Data Cleaning and Wilson Editing

Data preprocessing is a task that improves the quality of the data before they are used to their analysis [15][9]. Due to the fact that dataset has many missing values and based on type of variable, we have decided to use the attributes mean and mode to fill them. This means that whether the variable is numerical, we will use mean and we will use mode on nominal variables. For example, ICFES is a numeric variable then we will use mean by class. Using this approach, we have replaced 201 patterns by meand and mode.

Another preprocessing technique used in this study is Wilson Editing. This technique eliminates all patterns that are misclassified using KNN algorithm where the value of K is 3. Wilson Editing allows to increment the classifier's performance and to eliminate outliers. Wilson editing algorithm is shown below [10][8].

Initialization;
$S \leftarrow X$
for $x \in X$ **do**
  if *it is misclassified using the KNN rule with prototypes in* $X - \{x_i\}$ **then**
    $S \leftarrow S - \{x_i\}$;
  end
end

Algorithm 1: Wilson Editing Algorithm

3.3 Gamma classifier

Gamma classifier is an unconventional algorithm which is based on alpha and beta operators and its operation is completely based on binary representation into the modified Johnson Moebius.[13][12]

Definition 1: The operators Alpha and Betha are defined in table 2, given the sets A={0,1} and B={0,1,2}.

Table 2: Alpha and Beta operators

(a) Alpha Operator

$\alpha : A \times B \rightarrow B$		
x	y	$\alpha(x,y)$
0	0	1
0	1	0
1	0	2
1	1	1

(b) Beta Operator

$\beta : B \times A \rightarrow A$		
x	y	$\beta(x,y)$
0	0	0
0	1	0
1	0	0
1	1	1
2	0	1
2	1	1

1. Code the fundamental set into the code Modified Johnson Moebius and to obtain the value of e_m for each component.

$$e_m = \bigvee_{i=1}^{\rho} x_j^i \tag{1}$$

2. Compute the stop parameter.

$$\rho = \bigvee_{j=1}^{n} e_m(j) \tag{2}$$

3. Code the test pattern y with the Modified Johnson Moebius. If y_j is greater than e_m, Gamma operator will use y_j instead of e_m.
4. Transform the index of all patterns into two indices, one for their class and another for their position in the class.
5. Define the weight of each dimension Suggested empirical values are detailed below:

   a Within the range [1.5, 2] to features that are separable.
   b Within the range (0,0.5] to features that are not separable.

6. Initialize $\theta = 0$
7. Compute $\gamma_g(x_j^{i\omega}, y_j, \theta)$ for each component of the fundamental patterns in each class, according to:

$$\gamma_g(x_j^{i\omega}, y_j, \theta) = \begin{cases} 1 & \text{if} \quad m - u_\beta[\alpha(x,y) mod 2] \leq \theta \\ 0 & \quad Otherwise \end{cases} \tag{3}$$

8. Compute the weighted sum c_i for each class, as follows:

$$c_i = \frac{\sum_{\omega=1}^{k_i} \sum_{j=1}^{n} \gamma_g(x_j^{i\omega}, y_j, \theta)}{k_j} \tag{4}$$

9. If there is more than a unique maximum among c_i, increment θ by 1 and repeat steps 7 and 8 until to find a unique maximum or the stop parameter is fulfilled with the condition $\theta \leq \rho$.
10. If there only one maximum, assign y to the class which correspond such maximum.

$$C_y = C_j such that \bigvee_{i=1}^{m} c_i = c_j \tag{5}$$

11. Otherwise, assign y to the class of the first maximum.

4 Experimental Results

The aim of this study is to determine with higher precision whether a student will pass or not Mathematics course and overcome the results presented in [4] using the method of Gamma classifier and in accordance with the obtained results, we have considered seven classifiers provided by a system which has a collection of machine learning used basically to data mining task called WEKA (Acronym for Waikato Environment for Knowledge Analysis)[16] to compare the results. For our purpose, we used WEKA 3.6.10 version software and the algorithms are listed below.

1. Naive Bayes
2. Bayes Net
3. Support Vector Machines
4. Simple Logistic
5. Logistic
6. KNN where K=3
7. Tree Decision: J48 algorithm

Table 3: Confusion Matrix

Predicted class	**True class**	
	Positive	**Negative**
Positive	True Positive (TP)	False Positive (FP)
Negative	False Negative (FN)	True Negative (TN)

The elements can be detailed as follows [2]:

- True Positive (TP) is the number of correct predictions that an instance is positive.
- False Positive (FP) is the number of incorrect predictions that an instance is negative.
- False Negative (FN) is the number of incorrect predictions that an instance is positive.
- True Negative (TN) is the number of correct predictions that an instance is negative.

Using the elements described above, we can obtain the following measures:

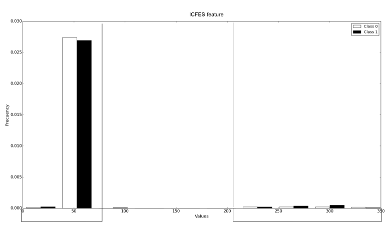
$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$precision = \frac{TP}{TP + FP} \tag{7}$$

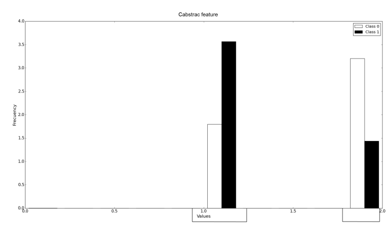
$$recall = \frac{TP}{TP + FN} \tag{8}$$

$$F - Measure = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \tag{9}$$

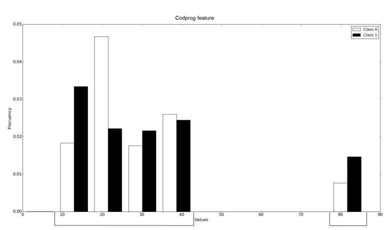
First, we applied data cleaning and Wilson Editing to dataset provided by Technological University of Pereira [4]. Then, based on Gamma algorithm, a combination of weights were tested to each feature. The combination of weights is arranged according to the histograms and were selected when Gamma classifier was run and provided the best results. Figure 1 shows the feature's histograms.



(a) ICFES



(b) Cabstrac



(c) CodProg

Fig. 1: Histograms of each feature used in this study

Based on the histograms, we can see that the classes are not separated, this means that the features have almost the same values, thus we have decided to assign a weight of 0.1 instead of 0. Table 4 details a final combination of these weights for each feature with the aim of obtaining better results.

Table 4: Proposed weights

Variables	**Weight**
ICFES	0.5
CodProg	0.1
Cabstract	0.1

The following step is to validate Gamma classifier a stratified k-fold cross validation was used in this study. Finally,Gamma classifier was run and it was compared with the algorithms provided by WEKA [16].The results of this experiment are detailed in table 5.

Table 5: Performance Measures

Classifier	**% Acc.**	**Precision**		**Recall**		**F-measure**	
		0	1	0	1	0	1
KNN	94.48	0.932	0.885	0.913	0.91	0.922	0.897
J48	91.14	0.893	0.836	0.875	0.859	0.884	0.847
Bayes Net	81.78	0.81	0.832	0.892	0.718	0.849	0.771
Gamma	**77.93**	**0.8576**	**0.7629**	**0.7376**	**0.8353**	**0.7931**	**0.7634**
SMO	77.60	0.857	0.698	0.732	0.835	0.789	0.761
Simple Logistic	77.26	0.856	0.694	0.726	0.835	0.785	0.758
Logistic	76.93	0.855	0.689	0.72	0.835	0.782	0.755
Naive Bayes	76.92	0.847	0.693	0.729	0.824	0.784	0.753

Some works have been made by Technological University of Pereira using the same dataset that is being used in this research and the purpose of the table 6 is to show the results obtained by them [4] [5].

Table 6: A comparison between Gamma Classifier and Logistic Multiple Regresion

Classifier	**% Performance**
Gamma classifier	**77.93%**
Logistic Multiple Regresion	70.40%
Logistic Multiple Regresion	61.8%

A unconventional classifier to determine whether a student will pass or not Mathematics course was applied and in order to compare its result, we have considered to use some algorithm provided by WEKA [16] according with the aim of the study. The results for this experiment shows that KNN produced the best result followed by J48 and Bayes Net with a performance of 94.4816%,91.1371% and 81.7726%, respectively. However, Gamma classifier produced a result of 77.9264% and it is evident that it overcame the results published in [4] (Details are given in table 6) as well as got better result than SMO with a classification rate of 77.592% followed by Simple Logistic with a classification rate of 77.2575%, Logistic with 76.9263% and finally, Naive Bayes with a classification rate of 76.9231%.

With regards to the results in this experiment, Gamma classifier can be used like a border to determine algorithms with high performance and algorithms competitive. It can be seen from the table 3 that Gamma classifier has competitive results in front of some algorithms provided by WEKA [16] and it can be used like a solution for determining whether a student will pass or not the Mathematics course.

5 Conclusions

This study presents a unconventional algorithm called Gamma classifier and two data preprocessing: Data cleaning and Wilson Editing approach for determining whether student will pass or not Mathematics course. The result obtained is competitive with a classification rate of 77.9264% and outperformed the results published by [4] with a classification rate of 70.4% and [5] with a classification rate of 61.8/

However, some considerations should be taken to determine more accurately the situation of a student in the Mathematics course. Firstly, adding social, academic, health risk and personal variables which can be very useful to obtain better result. Secondly, applying techniques for creating automatic weights. Finally, balancing methods for unbalanced dataset should be applied to improve the rate of classification.

References

1. Z. Abdullah, T. Herawan, N. Ahmad, and M. M. Deris. Mining significant association rules from educational data using critical relative support approach. *Procedia Soc. Behav. Sci*, 28:97–101, 2011.
2. M. Acikkar and M. F. Akay. Support vector machines for predicting the admission decision of a candidate to the school of physical education and sports at cukurova university. *Expert Syst. Appl.*, 36(3):7228–7233, 2009.
3. B. Baradwaj and S. Pal. Mining educational data to analyze student's performance. *Int. J. of Adv. Comput. Sci. Appl*, 2(6):63–69, 2012.
4. P. Carvajal, J. C. Mosquera, and I. Artamonova. Modelos de predicción del rendimiento académico en matemáticas i en la universidad tecnológica de pereira. *Sci. Tech.*, 43:258–263, 2009.

5. P. Carvajal, J. C. Mosquera, and I. Artamonova. Rendimiento en matemáticas i en la universidad tecnológica de pereira. *Sci. Tech*, 41:379–383, 2009.
6. M. Chalaris, I. Chalaris, C. Skourlas, and A. Tsolakidis. Extraction of rules based on students' questionnaires. In *2nd International Conference on Integrated Information*, volume 73, pages 510–517, 2013.
7. D. Delen. A comparative analysis of machine learning techniques for student retention management. *Decis. Support Syst*, 49(4):498–506, 2010.
8. C. F. Eick, N. Zeidat, and R. Vilalta. Using representative-based clustering for nearest neighbor dataset editing. In *IEEE International Conference on Data Mining (ICDM-04)*, pages 375–378, 2004.
9. C. Hernández G. and J. Rodríguez R. Preprocesamiento de datos estructurados. *Vínculos*, 4(2):27–48, 2008.
10. S. L. Donghai Guan, Weiwei Yuan, and Young-Koo Lee. Semi-supervised nearest neighbor editing. In *IEEE International Joint Conference on Neural Networks (IJCNN 2008)*, volume 8, pages 1183–1187, 2008.
11. H. Guruler, A. Istanbullu, and M. Karahasan. A new student performance analysing system using knowledge discovery in higher educational databases. *Comput. Educ*, 55(1):247–254, 2010.
12. C. López-Martín, I. López-Yáñez, and C. Yáñez-Márquez. Application of gamma classifier to development effort prediction of software projects. *Nat. Sci. Publ.*, 418(3):411–418, 2012.
13. I. López-Yáñez. Clasificador automático de alto desempeño (in spanish). Master's thesis, Center for Computing Research, National Polytechnics Institute, 2007.
14. I. Lykourentzou, I. Giannoukos, V. Nikolopoulos, and and V. Loumos G. Mpardis. Dropout prediction in e-learning courses through the combination of machine learning techniques. *Comput. Educ*, 53(3):950–965, 2009.
15. D. F. Nettleton. Data mining of social networks represented as graphs. *Comput. Sci. Rev*, 7:134, 2013.
16. I. H. Witten and E. Frank. *Data Mining Practical Machine Learning Tools and Techniques.* Morgand Kaufmann Publishers, 2nd edition, 2005.