Journal of Physics:
Conference Series

PURPOSE-LED PUBLISHING™

PAPER • OPEN ACCESS

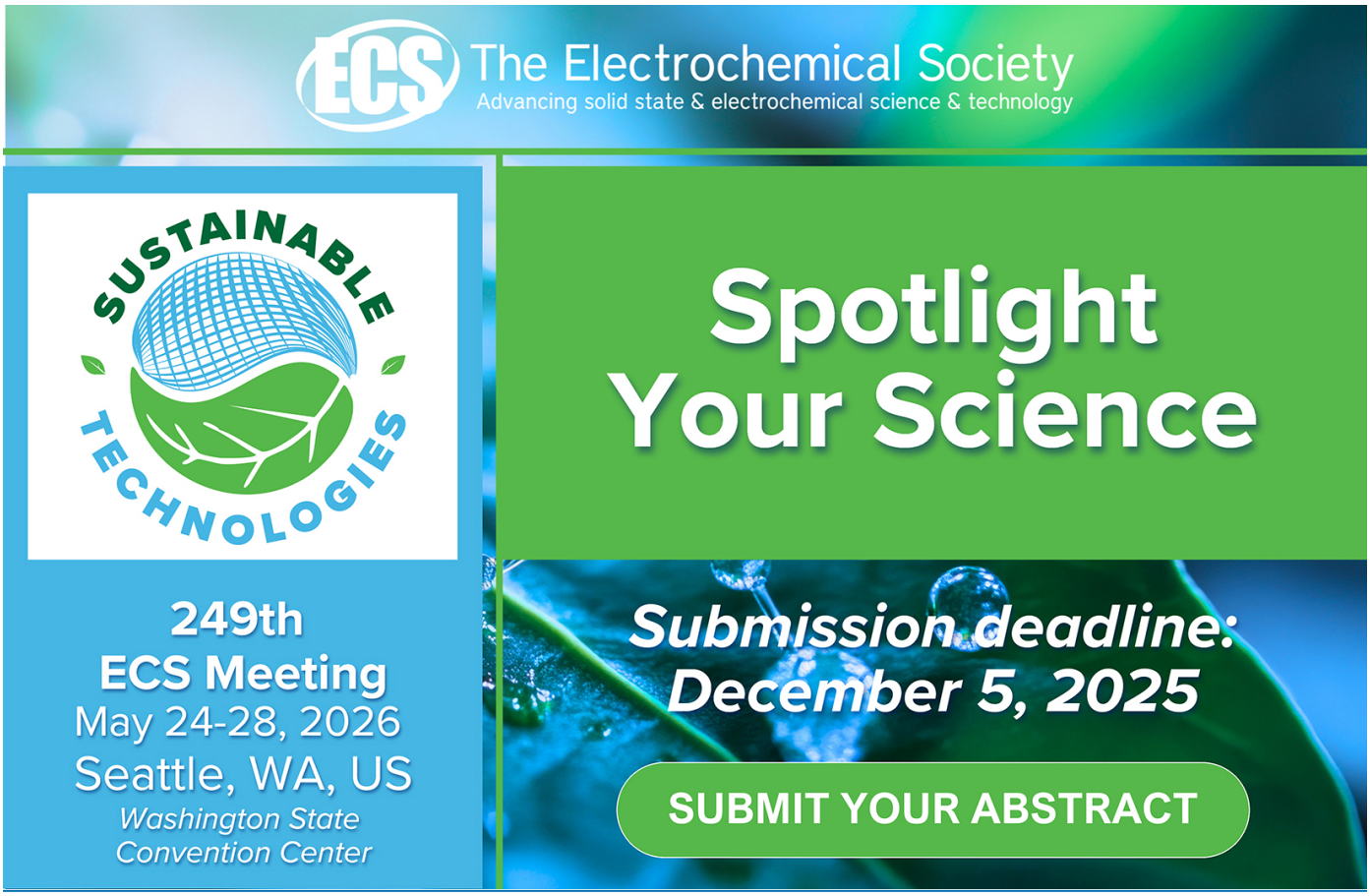
Development of software to predict academic performance using data mining techniques and tools

To cite this article: B Y Chindoy Chasoy *et al* 2020 *J. Phys.: Conf. Ser.* **1708** 012037

View the article online for updates and enhancements.

You may also like

- Level of development of mathematical logical thinking in the students of the agricultural areas of the Universidad Francisco de Paula Santander, Ocaña, Colombia
C L Garcia Quintero and C M Durán Chinchilla
- Applications of the Poisson and diffusion equations to materials science
C Nolasco, N J Jácome and N A Hurtado-Lugo
- Mathematical model of the firing process in Ladrillera Ocaña, Colombia
C Nolasco, G Gurrero Gómez Jácome and J A Gómez Campero



This content was downloaded from IP address 8.243.64.131 on 10/10/2025 at 14:18

Development of software to predict academic performance using data mining techniques and tools

B Y Chindoy Chasoy[1], **K Y Diaz Pedroza**[1], **and A A Rosado Gómez**[1]
1 Grupo de Investigación en Tecnología y Desarrollo en Ingenierías, Universidad Francisco de Paula Santander, Seccional Ocaña, Colombia

E-mail: bychindoyc@ufpso.edu.co, aarosadog@ufpso.edu.co

Abstract. Student desertion due to academic performance is understood as the interruption that a student makes voluntarily or by warning of his academic continuity by not achieving an acceptable accumulated average that keeps him active within his curriculum. The objective of this research was to develop a software that allows from the extraction of the data to the prediction of the academic performance of the students of the systems engineering program of the Universidad Francisco de Paula Santander, Ocaña, Colombia, in order to create strategies to reduce student dropout rates within the career. The methodology used was qualitative research with a descriptive approach. The life cycle in the development of the software whose name is Universidad Francisco de Paula Santander, Ocaña, Colombia, desertion was followed with the merger between open up and the knowledge extraction process. The type of learning used was supervised, obtaining that the highest percentage of correctly classified instances was achieved with algorithm randomizable filtered classifiers as well as J48 with a percentage of 92.6307% and 79.3185%, respectively. In addition, this software is applicable for any area in science and academia, since they worked with standard variables of which, the attributes with the highest incidence in low academic performance are, student status, academic sanction, the assessment of the proof of status, age, gender, social status and place of origin.

1. Introduction

On the university campus throughout history, student dropout has been present in the different academic programs related to academic performance and the system in general [1], understanding that academic performance is a variable of high impact on student dropout according to the Ministerio de Educación Nacional (MEN), Colombia. One factor that public and private higher education institutions (HEIs) have in common is that the difficulty that students present in learning when interpreting a text seriously affects the academic area, leading said students to imminent dropout [2]. This is why the educational system has established different strategies to try to reduce the numbers of dropout students, such as the creation of recreation and sports programs, the monitoring present to emphasize the shortcomings that may occur in some subjects, among other.

With the appearance of terms such as data mining and data analysis, part of the efforts that were previously tedious with the use of these elements both in the educational environment and in different areas of the market, have given the ease, speed, and easy interpretation of large volumes of data, of which previously there was no value. Obtaining in this way, rules and patterns [3] that allow much more detailed and precise decision making, reducing the waste of time in mechanical tasks [4], due to the use of software tools that are developed in order to show what is not obvious in naked eye interpretation [5].

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

In this case, it is necessary to mention that, taking as a reference to different techniques of data mining and knowledge in software development, the result is an information system that follows the knowledge discovery in databases (KDD), which allows the detection of students with low academic performance, being this one of the variables that most influences student dropout [1], and due to the creation of this software it can be predicted six (6) months in advance the average each student will have at the end of their academic semester, thus contributing to the generation of different strategies for each student through the curriculum, depending on the difficulty that is present.

2. Methodology

In the development of an investigation, the use of a methodology that takes the course of the same is of vital importance. For this case, the qualitative descriptive methodology was used, taking as a population the students of the systems engineering career of the Universidad Francisco de Paula Santander, Seccional Ocaña (UFPSO), Colombia, and it was divided into four basic phases; phase 1, knowledge of the environment, is concerned with understanding the problem in-depth. Phase 2, development of the KDD procedure, each of the activities related to the extraction, treatment, transformation, generation, and validation of knowledge. Phase 3, construction of the data mining tool, follow the analysis and development of the application, based on the activities of KDD and with the algorithms that produce the best performance, and finally phase 4, evaluation and interpretation of the operation of the tool and the results obtained with it.

3. Development of the phases

Below, it is described what was done in each phase, indicating the resources that were used to fulfill the proposed activities.

3.1. Knowledge of the environment

For the development of this phase, visits were made to the systems engineering curriculum in order to obtain the necessary information on the influence that low academic performance has on student dropout and in this way their support was obtained in the development of the investigation, therefore, this unit managed to obtain all the systems engineering data from the admissions, registry and control office of the UFPSO.

3.2. Development of knowledge discovery in databases

With the KDD procedure, in the development of the research, the set of activities that must be developed in a data mining process were provided, which goes from the selection of the data to the interpretation of the learning obtained, which allows identifying valid patterns, with novelty, very useful and easy to interpret from the data provided [6].

3.2.1. Data selection. In this phase, the necessary data for the research on which the discovery of the information is developed are selected and analyzed, where two important elements are taken into account, such as, on the one hand, the source that supplies the data, which in this case is the admissions registry and control unit of the UFPSO, who are the administrators of the students' information within the institution and on the other hand is the establishment of the variables to be used within the investigation, for this the results of a previous investigation were taken as a basis where the variables were established to take into account in research on academic achievement, based on data mining. The following were chosen for this project: student_state, sanction, year_entry, current_semester, icfes_ valuation, age, gender, social_stratum, place_of origin department_of origin, college_of origin, occupation, father_job, mother_job, academic_scholarship, address, number _of family_members, displaced, indigenous population, conflict, educational level of the person in charge, number of brothers studying, average (class) [7].

3.2.2. Pre-processing. In this phase, the quality of the data is guaranteed through the discovery of the different "errors" in which according to [8] the idea is to eliminate elements such as imprecise values (School origin), the existence of synonyms, and incorrect values (occupation, father_job, mother_job, academic_scholarship), substring with invalid content for data type (Address), semi-empty tuples (number _of family_members, displaced, indigenous population, conflict) and the existence of a conflicting column (educational level of the person in charge), which took place to be eliminated within the present investigation.

3.2.3. Transformation. This phase allows the discretization of the data in the variables so that in this way a much more compact and readable data set can be obtained. For this reason, use is made of norms given by the state or the discretization formula in which the highest value is established as maximum, minimum the lowest value and the #bins the total amount of data in the attribute being submitted and defined by Equation (1) [9]:

$$\text{Interval} = \frac{\text{Maximo}-\text{Minimo}}{\#\text{bins}}. \quad (1)$$

3.2.4. Data mining. This is the longest phase within the KDD procedure, which is divided into two main parts: a selection of attributes and creation of the model based on the defined algorithms.

- Attribute selection with weka. In the selection of attributes, the filters provided by the Weka tool were taken into account, where algorithms are used. In this case were: J48, Naive Bayes and RandomizableFilteredClasiffier, Search methods: BestFirts, GreedyStepWise and Ranker and Evaluation methods: CfsSubsetEval, ClassifierSubsetEval, OneRAttributeEval and CorrelationAttributeEval, thus allowing to eliminate the attribute civil status, since it is the attribute that contributes the least to the class.
- Model creation. In the construction of the model, the most referenced algorithms were taken into account, for this perhaps the J48, the Naive Bayes and the randomizable filtered clasiffier [7], todos con el objetivo de predecir y pertenecientes al método de clasificación. As shown in Table 1, in the case of algorithm J48 (A1) the percentage of correctly classified instances is 79.3185% with a confidence factor of 0.51. The behavior of the Naive Bayes (A2) algorithm in the classification of instances is that 56.1807% of instances in the data set are correctly classified. In the case of RandomizableFilteredClasiffier (A3), the percentage of correctly classified instances is 92.6307%, thus being the algorithm with the best classification behavior in the creation of the model. The attributes that were measured to each algorithm (alt) are: correctly classified (cc), incorrectly classified (ic), kappa statistic (ks), mean absolute error (mae), mean squared error (mse), relative absolute error (rae), relative squared error (rse), number of instances (ni).

Table 1. Instance classification.

alt	cc	ic	ks	rae	rse	ni
A1	79.3180%	20.6815%	0.7274	35.6%	59.7 %	1262
A2	56.1807%	43.8193 %	0.4087	69.8%	86.1 %	1262
A3	92.6307%	7.3693 %	0.904	11.3%	32.8 %	1262

3.2.5. Model evaluation methods. In this research, three methods were applied when evaluating the effectiveness of a model in data mining, which are: simple validation, cross-validation, and split validation. In the first, a test data set and a training data set are randomly taken, thus obtaining the model estimation error. In the data set that was used in this validation, it turned out to be the randomizable filtered classifier the model with the smallest prediction error. In the second validation, a small portion of the totality of the data is used, in this case, it was 10% that is divided into n groups for the training set and n-1 for the test, of which J48 had a better performance with a confidence factor of 0.25 and in

the last Validation 60% is defined as training data and the rest as a test, at the end of this validation the algorithm with the highest number of correctly classified instances was J48 [10].

3.3. Construction of the data mining tools

In accordance with the above, and having clear the bases at the data mining level, the software was made that would meet the requirements of the client and / or user; and meet the main objective that is the prediction of the average of the students of the systems engineering study plan. To do this, it was necessary to take into account the type of software, the programming languages, the database manager, the software architecture, among others.

3.3.1. Programming language. For the development of the software the Java programming language was used; this type of language allows for faster, safer, and more reliable software. The decision to use this type of programming language was because it contains a set of libraries that incorporate functionalities that allow the KDD to be developed and additionally correctly adapted to the type of software requested by the user, standalone software, since it was only required for a specific computer and the execution did not require internet access. On the other hand, the integration of this programming language with the software provided by the University of Waikato Weka, allows new applications to integrate calls to libraries that perform data mining tasks since Weka works with the same programming language [11].

It is important to highlight that for the creation of the software it is necessary to take into account the database management system (DBMS) since, it is who allows to administer and manage the information contained in the databases. In this case, the manager was MariaDB, being one of the most popular systems in the world, admitting a relational database model, with which the institution works, thus allowing the creation of a more competitive and independent software from the DBMS. [12]. Layered architecture and the model-view-controller design pattern. These are the elements that were used to structure the software described below.

- Layered architecture. This type of architecture has three important layers, which are: the presentation layer, the logical layer, and finally the persistence layer [13], this being the architecture taken into account for software development. Based on the aforementioned, the first layer is the one that allows the user to interact directly and who receives and issues notifications of changes in the system. The second layer is the one in charge of protecting the direct access to the information and controlling it so that it transits from layer one to the last in a bidirectional way, and finally, the last layer is the one that allows the storage of the data.
- Model-view-controller. It is a standardized design pattern whose objective is to divide the software architecture into three parts, separating the data from the user interface and placing it as an intermediary a controller that communicates the model with the view [14]. The view contains the interfaces that the user sees. In the part of the controller, all the actions exercised by the user are housed, which entail a call to action to the model or view. The model represents the specific part where the information of the system operates, in this case, the packages of the facade, expert, strategies and data access object (DAO) [12], as can be seen in Figure 1. This type of architecture is the one that shows more in-depth the internal functioning of the software known as the business logic. This type of architecture is concerned with creating processes that ensure quality in the programs. In addition to supporting a high structure at work, in which three parts can be evidenced, the model that contains the logical part of the application and internally has four layers (DAO, strategies, expert, and facade), the controller that allows communication between the view and the model, and finally the view, which are the interfaces with which the end-user will interact [15].

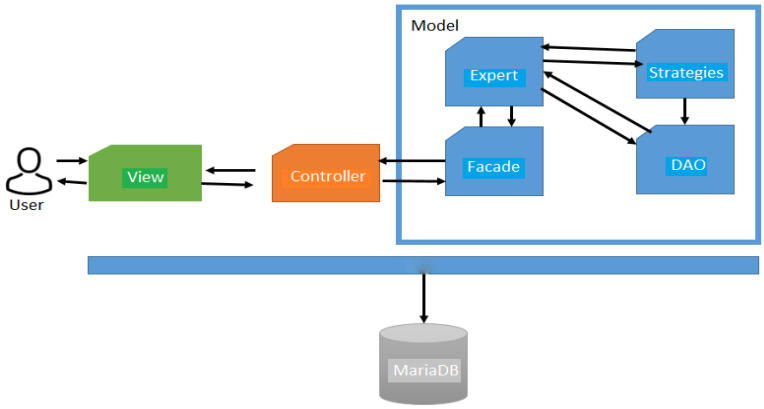
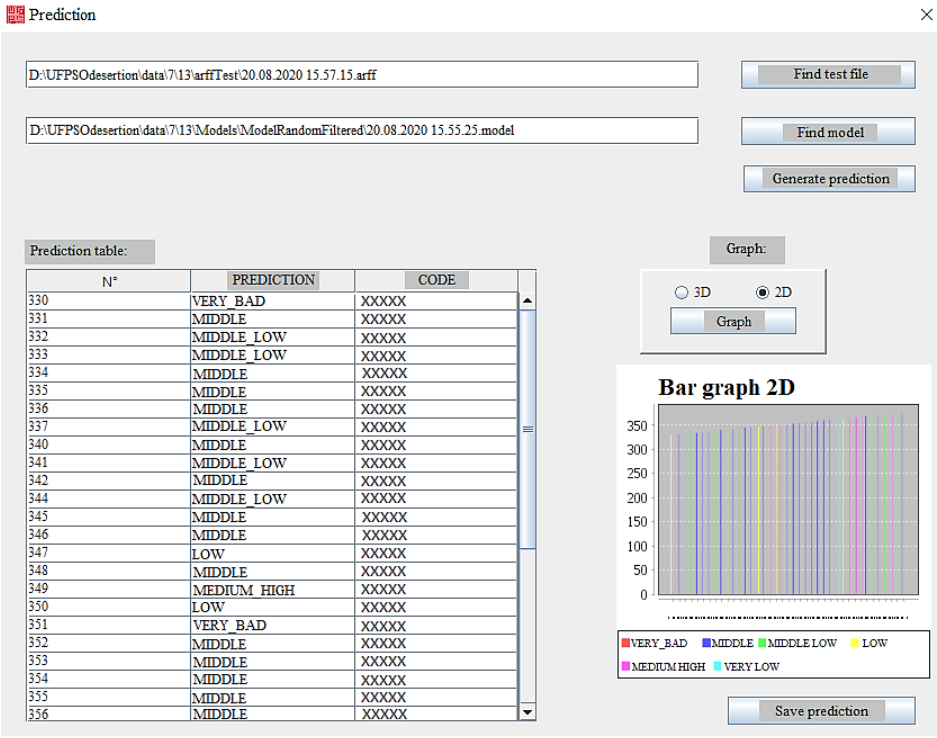


Figure 1. Representation of the model-view-controller design pattern within the project.

3.4. Evaluation and interpretation of the operation of the tool and the results obtained with it

Finally, the creation of the software fulfilled the main objective, the prediction of the average of the students in the systems engineering study plan, since when making a comparison between Weka and the UFPSO desertion application, the same results were obtained for which it fulfills with user requirements; and it shows that the tools implemented turned out to be adequate. Taking into account that academic performance is one of the factors that most influences student dropout and which was taken as a class within the software as can be seen in Figure 2, this being a view of the final result in an analysis made by the software developed between java and data mining elements, allowing the use of the class techniques offered by the KDD. Where, you can see the file path for the model, the path for the test data, and a table with the results when generating the prediction, which contains the student's code and the estimate in the prediction of their semester grade.



N°	PREDICTION	CODE
330	VERY_BAD	XXXXX
331	MIDDLE	XXXXX
332	MIDDLE_LOW	XXXXX
333	MIDDLE_LOW	XXXXX
334	MIDDLE	XXXXX
335	MIDDLE	XXXXX
336	MIDDLE	XXXXX
337	MIDDLE_LOW	XXXXX
340	MIDDLE	XXXXX
341	MIDDLE_LOW	XXXXX
342	MIDDLE	XXXXX
344	MIDDLE_LOW	XXXXX
345	MIDDLE	XXXXX
346	MIDDLE	XXXXX
347	LOW	XXXXX
348	MIDDLE	XXXXX
349	MEDIUM_HIGH	XXXXX
350	LOW	XXXXX
351	VERY_BAD	XXXXX
352	MIDDLE	XXXXX
353	MIDDLE	XXXXX
354	MIDDLE	XXXXX
355	MIDDLE	XXXXX
356	MIDDLE	XXXXX

Figure 2. Graphical representation of data mining and software.

4. Conclusions

With the development of the data mining tool for the detection of students who are at risk of having low academic performance, in the end, it is concluded that it is possible to identify students who are vulnerable to having low performance and thus provide them academic or psychological support, depending on the specific case and the variables that most influence each one. The most significant contribution of this tool is related to first-semester students, who, when identified, can better prepare themselves and in some way change the projection of their academic future.

In addition, it is possible to highlight that with the construction of this tool it is possible to automate the entire KDD process, which in turn contributes to streamline data mining tasks that provides a degree of automation to the decision-making process, predicting behaviors futures that at first glance went unnoticed by the people in charge of these tasks.

Finally, it is important to highlight that this type of technological development can be adapted or coupled in existing applications by incorporating the KDD within them; To achieve this, both the programming language used and the proposed software architecture allow the development of modules that can be integrated, without affecting the performance and stability of the system.

References

[1] Viloria A, et al. 2020 *Dropout-Permanence Analysis of University Students Using Data Mining* (India: International Conference on Intelligent Computing, Information and Control Systems)
[2] Reyes L, Carranza E C, Castro D P 2012 Causas psicosociales de la deserción universitaria *Revista Logos Ciencia & Tecnología* **4** 2145–2549
[3] Bedregal N, Aruquipa D, Cornejo V 2020 Data mining techniques to extract academic behavior profiles and predict university desertion *Revista Iberica de Sistemas e Tecnologias de Informacao* **2020** 592-604
[4] Rosado A 2016 Consolidation of institutional indicators using data warehouse *Revista Ingenio* **11(1)** 53-63
[5] Rosado A, Rico D Inteligencia de negocios: Estado del arte *Scientia Et Technica* **1** 321-326
[6] Quiroz N, Valencia C 2012 Aplicación del proceso de kdd en el contexto de bibliomining: El caso Elogim *Revista Interamericana de Bibliotecología* **35** 97-108
[7] Diaz K Y, Chindoy B M, Rosado A A 2019 Review of techniques, tools, algorithms and attributes for data mining used in student desertion *Journal of Physics: Conference Series* **1409** 012003:1-7
[8] Muñoz I, et al. 2019 *Calidad de Datos* (Bogotá: Ra-ma) pp194
[9] Rodríguez J 2010 *Fundamento de Minería de Datos* (Bogotá: Universidad Distrital Francisco José de Caldas)
[10] Fernández B 2018 *Validación Interna de Modelos Predictivos de Regresión Logística. Comando Validation (stata)* (Madrid: Universidad Complutense de Madrid)
[11] Martínez F, Hernández J P 2018 Técnicas de minería de datos con software libre para la detección de factores asociados al rendimiento *Revista de Estudio y Experiencias en Educación* **3** 135-145
[12] Chávez R D 2016 *Análisis del Framework Django para Implementar Aplicaciones Web con Base de Datos Mariadb y Metodologia de Desarrollo Scrum* (Ibarra: Universidad Técnica del Norte)
[13] Lagunes G, López I, Pelaéz G, Abud M A, Olivares B A 2015 Arquitectura de software de una aplicación móvil para desarrollar un sistema de identifiación por radiofrecuencia *Revista Electrónica de Computación, Informática Biomédica y Electrónica* **4(1)** 1-17
[14] Guzmán G, Tovar C 2013 *Implementación del Patrón de MVC para el Proceso de Selección de Personal* (Santo Domingo: Universidad Libre)
[15] Fernández Y, Díaz Y 2012 Patrón modelo- vista-controlador *Revista Telem@tica* **11** 47-57