International Journal of Engineering Education Vol. 34, No. 3, pp. 879–894, 2018
Printed in Great Britain

0949-149X/91 $3.00+0.00

Using Bayesian Networks for Learning Analytics in Engineering Education: A Case Study on Computer Science Dropout at UCLM*

CARMEN LACAVE and ANA I. MOLINA
Dpto. de Tecnologías y Sistemas de la Información, UCLM, Escuela Superior de Informática—Paseo de la Universidad, s/n, 13071—Ciudad Real, Spain. E-mail: {carmen.lacave, anaisabel.molina}@uclm.es

Student dropout in Engineering Education is an important problem which has been studied from different perspectives and using different techniques. This manuscript describes the methodology used to address this question in the context of *learning analytics*, using Bayesian networks because they provide adequate methods for the representation, interpretation and contextualization of data. The proposed approach is illustrated through the case study of the abandonment of Computer Science (CS) studies at the University of Castilla-La Mancha, which is close to 40%. To that end, several Bayesian networks were obtained from a database containing 363 records representing both academic and social data of the students enrolled in the CS degree during four courses. Then, these probabilistic models were interpreted and evaluated. The results obtained revealed that the great heterogeneity of the data studied did not allow to adjust the model accurately. However, the methodology described here can be taken as a reference for other works where a less heterogeneous database could be obtained, aimed at analysing student characteristics from a database.

Keywords: Learning analytics; Bayesian networks; Engineering dropout

1. Introduction

The abandonment of university studies is a reality that affects all universities, involving economic losses, social problems and possible psychological problems in the student and, consequently, it is one of the criteria used for evaluating higher education institutions, being a large concern to the education community and policy makers [1]. Therefore, it is very important to identify the students who tend to dropout a course from the beginning of their academic steps by giving them the extra support they need to avoid abandonment.

This problem affects significantly to Engineering studies and its explanation and prediction has been tackled from different perspectives and using different techniques since Tinto's first works [2], taken as the groundwork for recent research. There are studies based on the creation of surveys or questionnaires to be answered by the students and whose data are subsequently analysed using descriptive statistics. But in these cases, the prediction of students that are in "high risk" is often very difficult and time wasting [3]. Even if the identification was possible, these methods are not effective enough because it is often too late to avoid student dropout.

There is other perspective, based on *data mining* procedures [4], to extract relevant and interesting knowledge from data [5] that has been applied to predict both student performance [6] and student dropout [7–9]. There is a great variety of techniques, such as classification [3, 10], regression [11–13], decision trees [14–18], genetic algorithm [19] or a combination of several methods [20–22].

Nowadays, a multi-disciplinary approach has gained an increasing relevance in the use of big data analysis to inform decisions in higher education known as *Learning Analytics* [23–24], which is very related to *Educational Data Mining* [25], and that involves different areas related to machine learning and visualization. According to Clow [26], the most common definition of Learning Analytics is "the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments on which it occurs". Therefore, the prediction of dropout and retention in Engineering studies are key issues for Learning Analytics research [27, 28].

In this paper, Bayesian Networks (BN) are proposed to address this problem in the context of *learning analytics*, because they provide advantages over other models: they provide a compact way of representing knowledge and adequate methods for the interpretation and contextualization of data [29, 30]. These leads make easy for the researcher to understand the results without needing lot of statistical knowledge and, therefore, helping him or her to make decisions. In fact, BNs have been recommended to be used for modeling the complexities of higher education, since these models represent a "holistic", global approach to answering common institutional research questions [31] and are capable of handling the uncertainty in student-related data,

* Accepted 12 January 2018.

while also offering an intuitive, accessible modeling capability that supports the decision-making and policy-setting processes [32].

This proposal is illustrated through a case study to address the problem of abandonment of Computer Science (CS) studies at the University of Castilla-La Mancha (UCLM), which is close to 40%. The database was provided by the UCLM and it contained 2570 records with 28 fields corresponding to both academic and social data of the students enrolled in the CS degree in the two campuses in which these studies are taught (Albacete and Ciudad Real), during the courses from 2008–2009 to 2011–2012, including information on whether they had abandoned or not the degree. Having prepared the database, two kinds of Bayesian network models were learned, one based on classificatory models and other involving the probabilistic relations among all variables in the database, to know which features in the available data are the strongest predictors of university abandonment [6]. The models were compared by several evidence cases propagation; besides a *total abduction* algorithm on all variables was applied in each case to identify the most probable profile for the student that leaves CS studies.

Therefore, the objective of this paper is twofold:

- On one hand, to build a model which allows to find out the dependence relations among the enrolment data of university students that could explain or shed new light on what motivate them to leave CS studies at the UCLM.
- On the other, to propose a methodology in the context of Learning Analytics based on the use of Bayesian networks, that could be applied to similar problems related with the acquisition of information from databases.

The paper is structured as follows: Section 2 presents the theoretical concepts on BN; Section 3 shows how BN can be used in the context of Learning Analytics; Section 4 illustrates the ideas described in the previous sections through a real case study; Section 5 discusses the main results; Section 6 reviews related works and Section 7 extracts the main conclusions of this work.

2. Bayesian networks

Bayesian networks are graphical models that use probability as a measure of uncertainty and they have been successfully applied to many real-world domains [33, 34]. Before introducing the theoretical concepts needed to understand these models, some background on graphs is reviewed.

2.1 Graph concepts

A *Directed Acyclic Graph* (DAG) is a directed graph without loops. This means that if there is an arc from node u to node v there must be no other directed path from v to u in the graph. In a DAG, nodes are related by three kinds of connections (Fig. 1) that describe, in an intuitive way, the dependence and independence relationships in a set of variables [35] thanks to the *d-separation* criterion [36]. Dependence between any two variables (or lack thereof) can be thought as the ability (or inability) of one of those variables to influence the other. Any of the two dependent variables respond together to a change in either of them, no matter what the direction of that response be. This influence (or lack of it) can be visualized as a communication channel between those variables. This channel can be open or closed. It is closed between two independent variables and open between dependent ones [37]. Those connections are the following:

- *Serial*, also known as *chain* or *linear*, illustrated in Fig. 1(a). In this kind of connection, A and C are dependent if the state of B is not known; once it is known, A and C become independent. The state of C is influenced only by the state of B, and no change in A is carried over to C. Therefore, it is said that B blocks the influence between A and C and then, A and C are independent given B.
- *Diverging*, shown in Fig. 1(b). In this case, B and C are dependent if the state of A is not known. If the state of A is known, B and C become independent. This means that if the common influence A is unknown, observing B changes the probability of C and vice versa. Once A is observed, it blocks the influence among B and C and so, B and C are independent given A.
- *Converging*, described in Fig. 1(c). In this case, A and B are independent if no observation is made on the common child C. If the state of C is unknown, it blocks the influence among A and B and they become independent. Once the common child is known, the influence flows from A to C and vice versa so that observing one of the parents (A or C) will explain away the other.

2.2 Definition of Bayesian network

A Bayesian network is a probabilistic graphical model that allows to represent probabilistic depen-

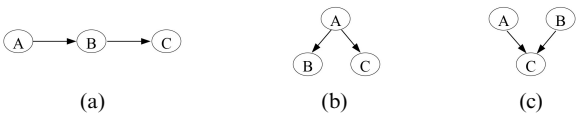

Fig. 1. Examples of the three possible connections in a DAG: (a) serial; (b) diverging; and (c) converging.

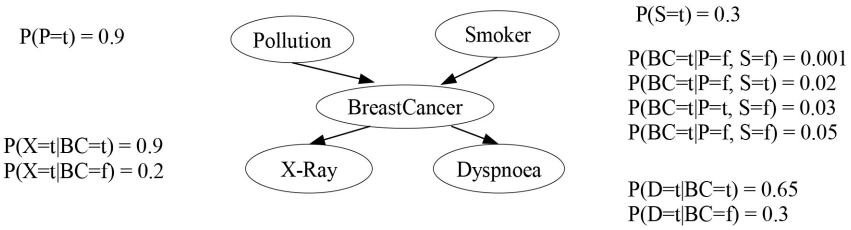

Fig. 2. Example of a Bayesian network representing the causes, symptoms and signs of breast cancer together the conditional probabilities of each node. Observe that to define probabilities, the names of variables have been abbreviated: P for Pollution, BC for Breast Cancer, S for Smoker, X for X-Ray and D for Dyspnoea. Moreover, all variables have been considered binaries with values true (t) or false (f).

dence and independence relations between a collection of data. It encodes a joint probability distribution over a set of random variables[1] $X = \{X_1, \ldots, X_n\}$ of a problem domain and it is defined in terms of two components [38]:

- *Qualitative component*: a DAG, where each node represents a variable and each arc between two variables indicates the existence of a probabilistic dependency between them. Variables can be discrete or continuous but in this work, we only consider the discrete case.
- *Quantitative component*: a conditional distribution $p(x_i \mid pa(x_i))$ for each variable $X_i, i = 1, \ldots, n$ conditioned on its parents in the graph, denoted as $pa(x_i)$. If the node has not any parent in the graph, its distribution is defined by its prior probability $p(x_i)$.

Considering the independencies represented by the network structure, the set of conditional probability distributions specifies a multiplicative factorization of the joint probability[2] distribution over X [36, 39], as Equation (1) shows.

$$p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i \mid pa(x_i)). \qquad (1)$$

Fig. 2 shows an example of a Bayesian network with five binary nodes representing a very simplified domain for diagnosing breast cancer (*Breast Cancer*) through its causes, as living in environments with high level of pollution (*Pollution*) and being a smoker (*Smoker*), one symptom, dyspnoea (*Dyspnoea*), and one sign, the results of an X-ray test (*X-ray*). From a mathematical or abstract point of view, BNs do not enforce the causal arc direction, although in this example every edge represents a causal relation among two nodes, defined by the probability of each node given its parents. For example, $P(BC = t \mid P = f, S = f) = 0.001$ indicates that the probability of a patient having breast cancer (BC = *true*), knowing that there is no pollution (P = *false*) and that he or she is not a smoker (S = *false*), is 0.001. This value can also be interpreted as that 1 in 1000 people who do not smoke nor live in polluted environments suffer from lung cancer.

2.3 Reasoning in Bayesian networks

A BN can be used to reason about the situation it models by applying algorithms which are based on the Bayes' Theorem (hence its name). The reasoning process, also known as *inference*, can be performed in two ways [36]:

- **Evidence Propagation**, which consists in computing the *posterior* probability distribution of the variables of interest, i.e., the probability distribution over each unobserved variable given that the value taken by some other variables is known. Many inference algorithms have been developed to compute the posterior probability distributions for all variables [40–42]. Since the problem is NP-hard [43], approximate algorithms are employed [36, 44, 45] when the models are so complex that exact solutions cannot be provided. That distribution reflects the influence of evidence and so, this type of reasoning is often used in systems where a diagnosis or a prediction is desired.
  - ***Diagnosis*** is a task of identifying the most likely causes given a set of observations. A good example of this kind of reasoning is medical diagnosis, where the focus is placed on identifying diseases a patient may be suffering from and ordering them from the most likely ones to the least likely ones given test results. In the *Breast Cancer* example, a doctor may be interested in diagnosing the absence of breast cancer in a non-smoker patient suffering dyspnea. In this case, setting as evidence the value *false* on variable *Smoker* and the value *true* on variable *Dyspnea*, the computa-

[1] Capital letters are used to denote variables and lower cases are used to represent the values (also called *states*) of a variable.
[2] $P(x_1, \ldots, x_n)$ denotes the probability that variable X_1 has the value x_1.and... and variable X_n has the value x_n.

tion of $P(BC = false|S = f, D = t)$ is performed by the propagation of that evidence.
  - ***Prediction***, on the other hand, attempts to identify the most likely event given a set of observations. For example, in the same *Breast Cancer* example, if the doctor knows that a patient is a smoker, he/she could be interested in predicting the probability of having breast cancer, i.e., to compute $P(BC = t|S = t)$. Then, the propagation of the evidence defined by the *true* value on variable *Smoker* will provide the required probability. Diagnosis looks at the past and present to reason about the present while prediction looks at the past and present to reason about the future [37].
- **Abduction**, used to look for the configuration of a set of variables, called *explanation set*, that maximize the joint probability given the observed evidence [36]. The abduction process is called *total* or *partial* depending on whether the explanation set contains every not observed variables or only just a subset of them. This process can be generalized to find also the k most probable explanations [46] and it is used to explain evidence and to extract profiles.
  - ***Explanation*** of evidence consists of identifying the most likely causes given a set of observations. For example, in the *Breast Cancer* example, the doctor may be interested in identifying the reasons that explain that a non-smoker patient has a breast cancer. In this case, the objective should be to obtain the "photo" of the state of the whole system modelling breast cancer. Therefore, obtaining the configuration of the unobserved variables, *Pollution*, *X-Ray* and *Dyspnoea,* that has maximum probability may shed light to explain the target variable *Breast Cancer* given the available evidence on *Smoker*.
  - ***Profiling.*** Moreover, in terms of data analysis, abductive inference tries to find the most likely *profile* of individuals in a population, under certain conditions imposed by the observed variables [29]. For instance, in the *Breast Cancer* example, the doctor may be interested in defining the profile of breast cancer patients. In general, the profiles may be given by some (partial abduction) or all (total abduction) system variables.

2.4 Construction of a BN

A Bayesian network can be developed manually [37, 47], from the specific literature and with the help of experts in the domain to model; automatically, by applying learning algorithms [48] or by a mixture of both [49].

The **manual** process consists of two main stages: (a) building the structure of a network with the help of human experts, by selecting the variables and drawing causal links among nodes, and (b) introducing the corresponding conditional probability distributions, given by conditional probability tables (CPT) when variables are discrete. In general, the manual construction of BNs encounters several problems. Ideally, those CPTs should be obtained from objective data but, in practice, the lack of objective data often forces the knowledge engineer to obtain the CPTs from human experts' estimations. Unfortunately, subjective estimates are often inaccurate due to different biases [50–51] and it is necessary to fix them. On the other hand, when the graph of the network grows large—say several dozens of nodes—it is more and more difficult for the human expert to intuitively grasp the structure of the network and for the knowledge engineer to explain it.

The automatic process, also known as **learning**, consists in taking a database and applying one of the many algorithms that yield both the structure and the conditional probabilities. This method requires the database to be properly designed and data is thoroughly collected. Most software tools provide functionalities to build Bayesian network from databases, but OpenMarkov [52, 53] provide learning wizards very easy to use. Learning can be classified into two types:

- ***Structural*** learning, by which the structure of the Bayesian network is obtained, that is, the relations of dependence and independence between the variables involved. There are several algorithms to build the network graph from a database, depending on whether the structure is fixed or not.
  - In the first case, the most used fixed structures are Naïve Bayes (NB) [54] and TAN [55] because of their good results, mainly used in *classification* problems. Fig. 3 shows an example of both structures. Naïve Bayes graph (Fig. 3(a)) is an easy structure defined by one parent classification variable (C) with several children ($X_1, X_2, \ldots, X_n$), the predictor variables, which are conditional independent given their parent. The TAN structure (Fig. 3(b)) is similar to NB but it incorporates

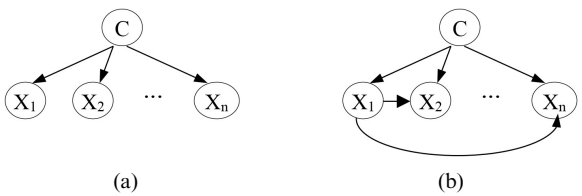

Fig. 3. (a) Naïve Bayes structure with one classification variable (C) and N predictor variables ($X_1, X_2, \ldots, X_n$). (b) TAN structure incorporating relations among features.

dependence relations among the predictor variables.
  - Alternatively, when the structure is not known, the best-known algorithms for obtaining it are K2 [56] and PC [57]. The algorithm K2 attempts to find an optimal network in terms of the likelihood of the database for each candidate network. In contrast, the PC algorithm tries to determine the structure of the network through statistical tests of independence. None of the methods is superior to other [38] and both operate with discrete variables; hence, continuous variables must be discretised [58].
- ***Parametric*** learning algorithms: provides the quantitative component of the Bayesian network defined by a given structure, i.e., the probability distribution associated with it. The parameters in this case are obtained by maximum likelihood estimation or by EM algorithm [59].

Finally, a Bayesian network can be developed from a **mixture** of manual and automatic construction, usually determining the structure manually by human experts and then, learning automatically the parameters from databases [60, 61].

Several software tools, such as Elvira [62, 63] or HUGIN [64, 65], allow to easily build Bayesian models, both manually and automatically, and provide libraries for performing both types of inference.

2.5 Model evaluation

The assessment of a BN's functionality is critical regardless of whether the model was built for description, classification, or prediction. The validity of a model is evaluated not only to ensure that the model describes the system of interest but also to perpetuate an ongoing, iterative process of critiquing and improving the model [32]. When data are available, comparing predicted values with actual values is the most straightforward way to discern the predictive accuracy of a model; unfortunately, most times this is not the case and some other methods must be used.

Models built using expert knowledge require different approaches to model validation. The experts participating in the elicitation should be asked to provide opinions of the final network's accuracy, and if data are available against which to compare, a BN can be evaluated using measures of predictive accuracy, deviations from expected value, and the extent to which predictions are calibrated (information reward) [66]. Other authors [67] recommend evaluating BNs through sensitivity analyses, in which the magnitude of the effects of changes in a network's structure or parameters are measured. When BNs are developed by a learning process, other measures of fit are used, such as the Bayesian Information Criterion (BIC) that also considers model parsimony [68] and Receiver Operating Characteristics curves (ROC) and confusion matrix that compare sensitivity against specificity [69]. A review of metrics related to predictive accuracy of models can be found in [70], although *cross-validation* is often used when there is no explicit set of test data. It is built upon the premise of partitioning data, so that a model can be learned from one data set (the training set) and the resulting derived model's predictive accuracy is evaluated against the remaining data (the testing set) [71]. There are many approaches to cross-validation in the literature, with general advice that the method chosen best represents the research goals and data characteristics while minimizing the trade-off between complexity and performance [72]. *K-fold cross-validation* involves randomly splitting cases into k equally sized partitions, cross-validating each partitioned sample across the remaining partitions, and then averaging predictive performance across all partitions. The main advantage of k-fold cross validation is that it allows the use of as much training data as possible while protecting against model over fit and providing measurements of predictive performance [32]. Additionally, when k is greater than two but also not too large, k-fold cross-validation at least partially addresses the "bias/variance" dilemma [73], in which minimization of potential bias and prediction error created by an inappropriate data split competes with the minimization of variance that is created by using a number of training sets to estimate model parameters. So that, it is recommended that k takes a value between 5 and 10 [72].

In addition to evaluating the predictive performance and sensitivity of a model, a final evaluative measure involves considering its complexity [32] as part of a holistic evaluation. Complexity can be measured by the number of variables, links and node states, and is typically used for comparing different models [70].

3. Learning analytics through Bayesian networks

Bayesian networks are a powerful tool to be used in the context of Learning Analytics by several reasons. First, as it has been said before, a Bayesian network provide a compact way of extracting knowledge from the application of learning methods, allowing to combine the knowledge given by a human expert with the data collected in a database, even having incomplete data. Thus, this facilitates the construction of models that adjusted to the

reality, avoiding the over adjustment that can be produced using data which represent only part of reality.

On the other hand, one of the main drawbacks of the methods commonly used in data mining is that they are difficult to interpret because they act as "black boxes", providing results without explanation. This trouble can lead to the lack of confidence by the user, which complicates decision making in any field, and to the inability of users to validate it [74]. However, one of the most important advantages of this Bayesian networks to be used in the context of Learning Analytics is that the structure of the associated DAG of a Bayesian network allows to understand, in an intuitive way, the relations of dependence and independence existing in a set of variables, even if the user does not have any knowledge on artificial intelligence [35]. If two variables are dependent, this relation can be represented by a path that connects these nodes; alternatively, if two variables are independent, there should be no way to join these nodes. In this context, the concept of dependence between variables is related to the concept of connection between nodes. Therefore, it is possible to find out, with no need of carrying out any numerical calculations, which variables are relevant or irrelevant for some other variable of interest (for instance, a prediction indicator). This process is also known as *relevance analysis* [38] and it is very useful to easily understand how information is transmitted in these models. We will illustrate through a toy example adapted from [36] how the relevance analysis is performed in Bayesian networks so that two variables are irrelevant if no information can be transmitted between them [38].

> Example (Sick or Love). Peter's parents just got home after spending a week away on a business trip when they see a letter in the mailbox informing them of their child's suspense on the University access test. They know that the only reason for Peter's failure is that he has not studied enough. Peter says to their parents that during the last week, during the examination days, he felt sick; however, when they are leaving home to go with Peter to the doctor, the neighbours congratulate them on how nice is the girl with whom they have seen Peter in and out of the house for the last week. Then, they decide to stay at home and avoid visiting the doctor.

The graph in Fig. 4 shows the five relevant variables and their causal relationships: being sick (*Sick*) and having a girlfriend (*Girlfriend*) can cause Peter does not study (*Not Study*) and this can cause the exam failure (*Exam Failure*). Moreover, if the neighbours see Peter dating a girl, they can tell it to Peter's parents (*Neighbours Tell*). Assuming the correctness of the example model (see [37] for a more detailed description about the manual development of causal Bayesian networks), the concept of *d-separation* introduced in section 2.1 explains how the existence of evidence on one variable might block the flow of information in a network and therefore, the network can be interpreted in the following way:

- The serial connection *Sick* → *Not Study* → *Exam Failure* reflects the fact that while the state of *Not Study* is not known, information about either *Sick* or *Exam Failure* will influence the belief on the state of the other variable. However, given that the state of *Not Study* is known, any information about the state of *Sick* will change the belief about *Exam Failure*, and vice versa. The serial connection *Girlfriend* → *Not Study* → *Exam Failure* can be interpreted in a similar way.
- The converging connection *Sick* → *Not Study* ← *Girlfriend* can be interpreted as follows: if no evidence is available about the state of *Not Study* then information about the state of *Sick* will not provide any derived information about the state of *Girlfriend*. In other words, being sick is not an indicator of having a girlfriend, and vice versa, i.e., they are independent. However, if evidence is available on *Not Study*, then information about the state of *Sick* will provide an explanation for the evidence that was received about the state of *Not Study*, and thus either confirm or dismiss *Girlfriend* as the cause of the evidence received for *Not Study*. For example, if Peter recognises that he has not studied, knowing that he is not sick increases the belief on he has a girlfriend. The opposite, of course, also holds true.
- In the diverging connection *Not Study* ← *Girlfriend* → *Neighbours Tell*, if the state of *Girlfriend* is not known, receiving information about *Neighbours Tell* will influence the belief about *Not Study* since having a girlfriend is a possible explanation for not having studied. The updated belief about the state of *Girlfriend* will in turn make update the belief about the state of *Neighbours Tell*. If the state of *Girlfriend* is known, the information received about the state of the either *Not Study* or *Neighbours Tell* is not going to change our belief about the state of *Girlfriend*, and consequently the belief about the other, yet unobserved, variable is not going to be updated.

Then, BN provide flexible methods of reasoning, and well-founded on probability theory statistically

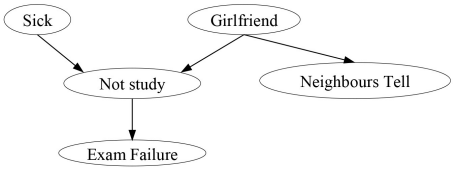

Fig. 4. The Bayesian network for the Sick or Love example.

robust enough, capable of providing meaningful results, predicting the value of unobserved variables and explaining the observed ones. Assuming that I is the indicator in which we are interested and E is a set of variables that can be controlled by the administration board, the prediction for the value of I given E can be obtained by computing the distribution $P(I|E)$ that would provide us the likelihood of each possible value of I given each possible configuration of E [38]. Besides, any variable of a BN can be either a source of information (if its value is observed) or object of inference (given the set of values that other variables in the network have taken). Reasoning will be then diagnostic or predictive depending on what evidence is available. The distinction between these two types of reasoning may blend in many real-life applications and it is often difficult to draw a line between them [37].

Several software packages provide easy interpretations of the modelled domain and the reasoning performed in the network. This is the case of Elvira [62, 63], which includes facilities to generate static and dynamic explanation for Bayesian networks.

To summarise, BNs comprise concepts and methods from different research areas involved in the Learning Analytics field, such as machine learning, artificial intelligence, information retrieval, statistics and visualization [25].

4. Related works

The analysis of educational databases [5] can help to the extraction of knowledge [77] in terms of certain characteristics such as academic performance [6, 78] or student dropout [7–9]. This analysis allows to model the behavior of dropouts, predict future dropouts, and identify patterns or profiles of students at risk, giving a chance to counsellors to advise and guide students into success. However, without the application of data mining techniques is very complicated to analyse the enormous amount of data available [79]. Different types of algorithms and techniques are used for information retrieval from educational databases [5].

Several works have tried to predict academic performance and the risk of dropping out of students [1, 80], applying different methods, such as classification [3, 10], regression [11–13], decision trees [14–18], genetic algorithm [19] or a combination of several methods [20–22] for their prediction in educational setting. To make such prediction, several attributes are used, such as academic, social, demographics, personal and family data.

In last years, the incorporation of BNs into educational and institutional research is gaining in popularity and application [31, 81–83]. BNs offer an excellent approach to dealing with the uncertainty inherent in educational research [84], also offering an intuitive, accessible modeling capability that supports the decision-making and policy-setting processes [85].

The study conducted by Kotsiantis et al. [86] is considered one of the pioneering studies that investigates the application of machine learning techniques for dropout prediction. The objective of this study was to identify the most appropriate learning algorithm for the prediction of students' dropout. Several experiments were carried out with data provided by the Hellenic Open University and it was concluded that the Naïve Bayes algorithm can be successfully used. In addition, a web based support tool was created to automatically recognize students with high probability of dropout. In [87], a model for predicting students' performance levels was proposed employing three machine learning algorithms: instance-based learning classifier, decision tree and Naïve Bayes. It was concluded that Naïve Bayes indeed performed better than any other machine learning algorithm. Fernández et al. [38] propose a methodology for analyzing performance indicators of higher education based on the use of Bayesian networks and apply the methodology for the particular case of the University of Almería. This study works with several indicators (student performance, average mark when admitted to university, etc.) and an analysis of relevance from the Bayesian network is obtained. Morales et al. [29] presented a study in which built a Bayesian network based on data of students from their university and they made an analysis of data by propagating probabilities and extracting profiles through abductive inference. In [88], the author uses Naïve Bayes classification algorithm to generate predictive models for engineering student's dropout management, based on the previous year student data. They collected the student academic data like High School grade, Senior Secondary grade, and student's family position, etc., to predict the student's performance and to find those students who need special attention. Sharabiani et al. [60] create a model using a database of the undergraduate engineering students at University of Illinois at Chicago. The specific objective of this model was to identify the students who might receive low grades and hence need extra help from the educational authorities. The suggested model was tested against the conventional models proposed in the literature and it performed better these in grade prediction.

Finally, some authors have compared the predictive performance of educational researches that use BNs to model with other techniques (mainly decision trees or neural networks) [60, 75]. Most of these studies tried to predict student success [89, 90] and dropout from online classes [20, 91]. Some of

them concluded that the accuracy of Bayesian networks was worse than the other two methods [20, 90, 91], while others found opposite results [89, 92]. On the other hand, there are authors like Nikolovski et al. [93] who state that the higher percentage of accuracy of any classifier algorithm is totally dependent on the quality of attributes and data model which are selected for the data collection.

In this paper, a different approach to address dropout prediction is presented and illustrated through a real case study. To that end, a database containing both academic and social data of the students enrolled in the CS degree during four courses is used to learn several Bayesian networks, both with fixed and unfixed structure. Then, these probabilistic models are interpreted, evaluated and compared, revealing some interesting results. The main innovations of the work are related, on one hand, to the use of unfixed learned structures as classification models, and, on the other, to the methodology used to interpret and evaluate them with the use of the Elvira program, because it provides specific facilities to interpret the results [75].

5. Case study

This section illustrates through a real case study how the automatic development of Bayesian network from a data collection can be used to address the analysis of CS dropout at University of Castilla-La Mancha.

5.1 Data collection

The UCLM office provided us with a database containing 2570 records representing both academic and social data of the students enrolled in the different CS degrees imparted at UCLM in Computer Engineering (ISI), Technical Engineering in Computer Management (ITIG), Technical Engineering in CS Systems (ITIS) and the Degree in Computer Engineering (GII), taught both in the Campus of Ciudad Real and in the Campus of Albacete of the UCLM, and including information about whether each student dropped out in some of the courses from 2008–2009 to 2011–2012. After eliminating those who moved to another CS career and those who come to the UCLM to study temporarily (Erasmus and Sicue/Seneca), the initial database was reduced to 363 records with 18 fields, corresponding to the data of students who dropped out in some academic year within the period 2008–2009 to 2011–2012. This stage of preparation of the database has been the most laborious, confirming what was revealed by other authors [61].

5.2 Automatic modelling of the Bayesian network

To address the main goal of our research work, a Bayesian network was developed to model the relationships among the data represented in the database in order to identify the different profiles of the student that leaves CS studies, and after, performing a relevant analysis.

First, one variable was defined for each of the eighteen field in the database. One of the main problems of BN is that the excess of granularity in the definition of the values of the variables can increase the computational complexity of the algorithms. Therefore, as a previous step, some variables with many values were simplified, such as those related to age, the city name of the home family, parent's studies and professions, and those related to the number of subjects enrolled, passed, validated and failed. For example, for the description of the family municipality it was simplified to represent only if the student came from a town or a province capital; the age values were grouped by intervals, as well as the number of subjects or the number of years that the student remains in the degree before leaving. In short, the variables considered in the analysis and their values were as follows:

- *CS_STUDIES (CS)*, for the names of the CS degrees imparted at UCLM, i.e., GII, II, ITIS, ITIG.
- *CAMPUS (CP)*, with values Albacete and Ciudad Real, the two campus names where CS studies are imparted.
- *SEX (S):* Male and Female.
- *AGE* (A) of the student when leaving studies, with 4 possible values: [20,25], [26,30], [31,40] and [41,50].
- *PROVINCE_FAM (PF)*: the name of the Spanish province where the student's parents live.
- *CITY (C)*: represents if the student family lives in a province capital or in a town.
- *KIND_ACCESS_UNIV (KAU)*, defining the way of access to Spanish University, i.e., Professional Training (PT), Validation of foreign studies (FOR), university entrance exam (UEE), access exam for people aged older than 25 (OLD25), Technical Engineers (TE) and Graduates (GR). The data base also contains the value UNKNOWN.
- *SUBKIND_ACCESS_UNIV (SAU):* different values depending on the kind of access to Spanish university, such as UEE_LOE, PT access, and so on.
- *MARK_ACCESS (MA)*, to define the access mark to University, if it applies. It has been discretized in the intervals [5,6), [6,7), [7,8), [8,9) and [9,10].
- *FATHER_STUDIES* (*FS*) and *MOTHER_-*

STUDIES (*MS*)*:* defines the studies level of the student's father and mother, respectively, with values NonApplies (when it is unknown), No Studies, Primary, Secondary, Higher.

- *FATHER_WORK* (*FW*) *and MOTHER_WORK* (*MW*): with values from 0 to 10 defining different professional levels, ranging from unemployed (0) to director of a large company (10), and a NonApplies value when the work is unknown.
- *ENROLLED_SUBJECTS (#ES)*: Total number of subjects enrolled in the academic years from 2008–09 to 2012–13 grouped in several intervals the intervals [0,5], [6,10], [11,15], [16,20], [21,25], [26,30], [31,35].
- *PASSED_SUBJECTS (#PS), FAILED_SUBJECTS (#FS)* and *VALIDATED_SUBJECTS (#VS):* Idem for passed, failed and validated subjects, respectively.
- *YEARS_NUMBER (YN)*: It represents the difference in years between the abandonment year and the first enrolment year. The database included students who were enrolled for the first time up to 13 years prior to the year they dropped out, so that some subsequent results may attract attention. The intervals for this variable are: [0,1], [2,3], [4,6] and [7,13].

Once the database was prepared according to the previous definition of variables, several BN were developed in a sequential process, depending on different objectives.

5.2.1 Fixed structure

Initially, the first aim of this work was to predict the abandoned studies according to the evidence known about a student, since during the years studied in this work 4 different CS curricula were offered. Therefore, a Naïve Bayes (NB) model was developed defining *CS_STUDIES* as the dependent variable and the rest as predictors. It was obtained the

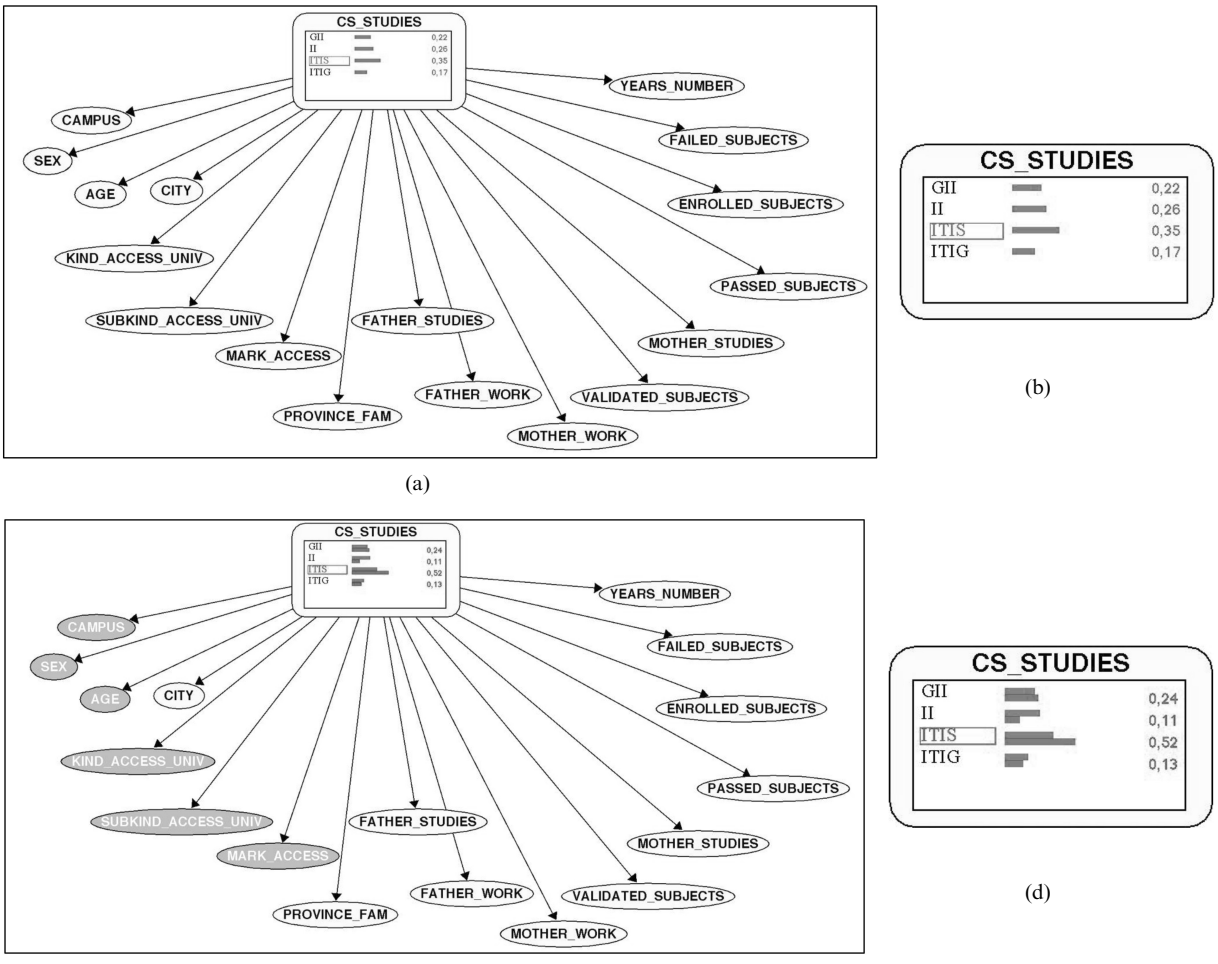

Fig. 5. (a) Naïve Bayes model representing all the variables of the database. (b) Zoomed variable *CS_STUDIES*. (c) NB network after propagating evidence on variables *CAMPUS*, *SEX*, *AGE*, *KIND_ACESS_UNIV*, *SUBKIND_ACCESS_UNIV*. The observed nodes are set in a darker colour. (d) Zoomed variable *CS_STUDIES* after evidence propagation. Both networks (a) and (b) have been obtained using the ELVIRA software.

BN shown in Fig. 5(a) which represents that the abandoned studies depend on the rest of variables of the database. Fig. 5(b) shows a zoom on CS_STUDIES revealing that, when no evidence is available, the most probable degree to be left is ITIS, with a probability of 0.35. In fact, the 35% of the records in the database correspond to ITIS. In this context, we reasoned in two ways: the first one was introducing certain evidence on some predictor variables to obtain which CS program is the most probable to be dropout. For example, for a student that has left CS studies, being a male, enrolled in Ciudad Real, he is 23 years old and he has accessed University through the most usual way, the Pre-registration in 1st year, after passing UEE, the evidence E = {*CP* = CiudadReal, *S* = male, *A* = [20–25], *KAU* = Pre-Registration, *SAU* = Selectividad, MA = 5} is set. Then, an evidence propagation process is performed to compute *P(CS_ESTUDIES|E)*, as shown in Figs. 5(c) and 5(d). It can also be observed how probabilities of every state of *CS_STUDIES* have changed after evidence propagation. Now, the probability that such example student left ITIS studies has increased from 0.35 (Fig. 5(b)) to 0.52 (Fig. 5(d)).

If the aim of the work was, for example, the prediction of the number of years that a student remains enrolled before leaving the studies, the dependent variable should be *YEARS_NUMBER*.

The other kind of reasoning was a total abduction process to identify the most probable profile of dropout student. According to such model, and with a probability of 0.0000113, the profile corresponded to a male among 20–25 years old, enrolled in Albacete campus, whose family resides in Albacete province, and he comes from a province capital, who accessed University through Pre-registration in 1st year and UEE_LOE with a mark of 5; his parents both had primary studies, hs father worked in a Level_7 job and his moher is unemployed. Moreover, the number of enrolled subjects were among 6 and 10; and the number of validated, passed and failed subjects were all among 0 and 5. Finally, he was only enrolled for one year. To validate the

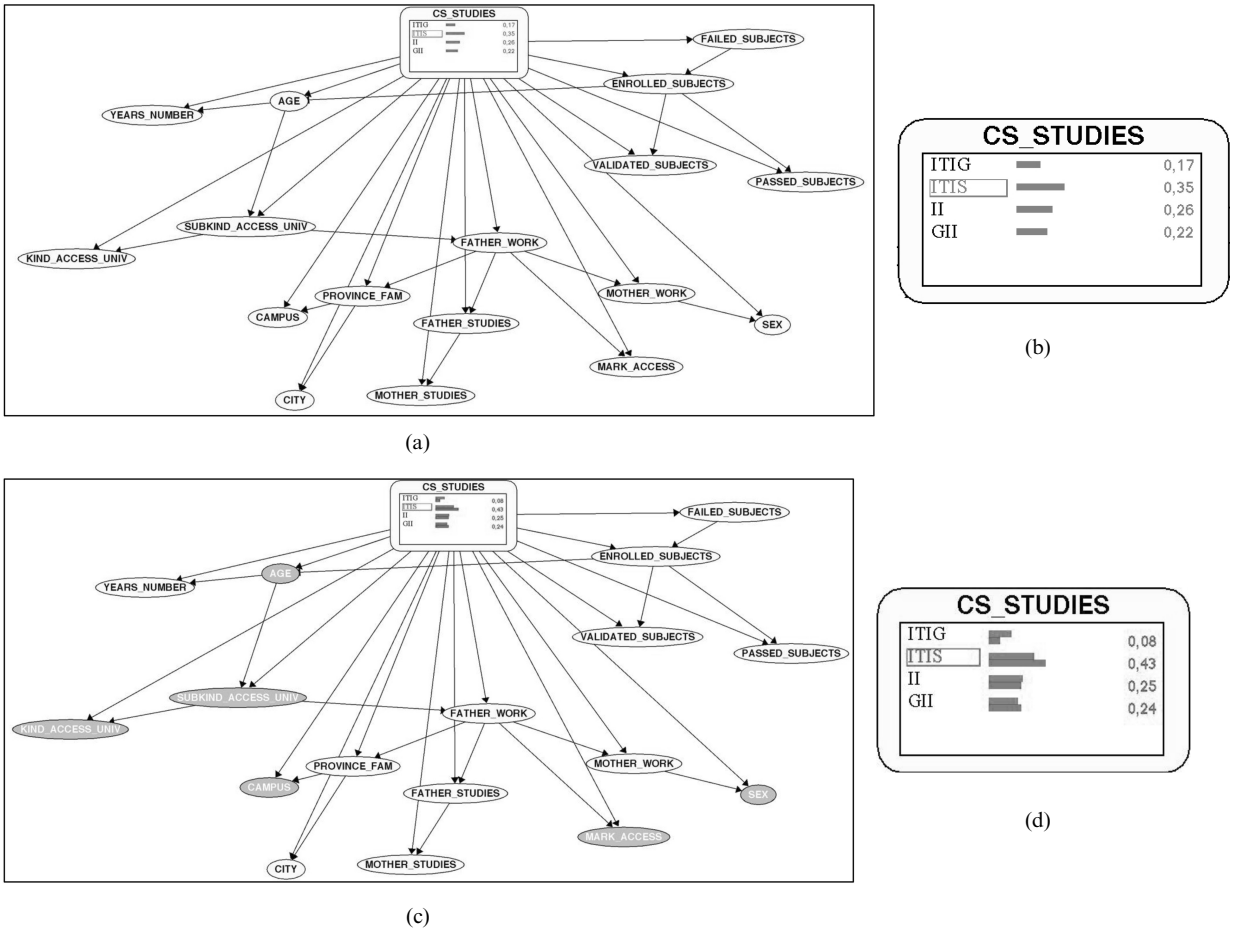

Fig. 6. (a) TAN model obtained with the Elvira program representing a classification problem on *CS_STUDIES* variable considering the dependences among the predictor variables. (b) Zoomed variable *CS_STUDIES*. (c) Network after propagating evidence on variables *CAMPUS*, *SEX*, *AGE*, *KIND_ACESS_UNIV*, *SUBKIND_ACCESS_UNIV*. (d) Zoomed variable *CS_STUDIES* after evidence propagation.

results, a search in the database was made to look for records with these values. However, there was no one corresponding to such profile. Consequently, this fact made us suspect that the model was not accurate enough.

After studying the graphical model, it was noticed that several predictor variables may not be completely independent. Therefore, a TAN algorithm was applied to represent those dependences among variables. The resulting BN is illustrated in Fig. 6(a), obtaining the same probabilities on *CS_STUDIES* (Fig. 6(b)). The learned model of Fig. 6(a) shows several reasonable dependences as, for instance, from *ENROLLED_SUBJECTS* to *VALIDATED_SUBJECTS* and *PASSED_SUBJECTS*, but there are others quite strange, such as the one from *FATHER_WORK* to *MARK_ACCESS*. Anyway, a similar reasoning process to that performed on the NB model, was made in this TAN model: the same evidence was propagated, obtaining other posterior probabilities on the unobserved variables (Fig. 6(c)). For example, for the findings as those of Fig. 5(c), $P(CS\_STUDIES=GII|E)$ is significantly different to the values obtained in the NB model (Fig. 6(d)). Also, a total abduction process was performed to get the most probable configuration of all variables and the obtained profile was equals to that found in the NB model except that in this case the father is also unemployed, the number of failed subjects is among 6 and 10, and the profile probability is a bit higher, 0.00007. Again, the database did not contain any record representing the most probable profile.

Because of such differences among the obtained results, considering also the many dependences among the predictor variables, and that in the database there was no record representing any of the two profiles, it was decided to build a prediction model but without imposing any fixed structure to the learning algorithm.

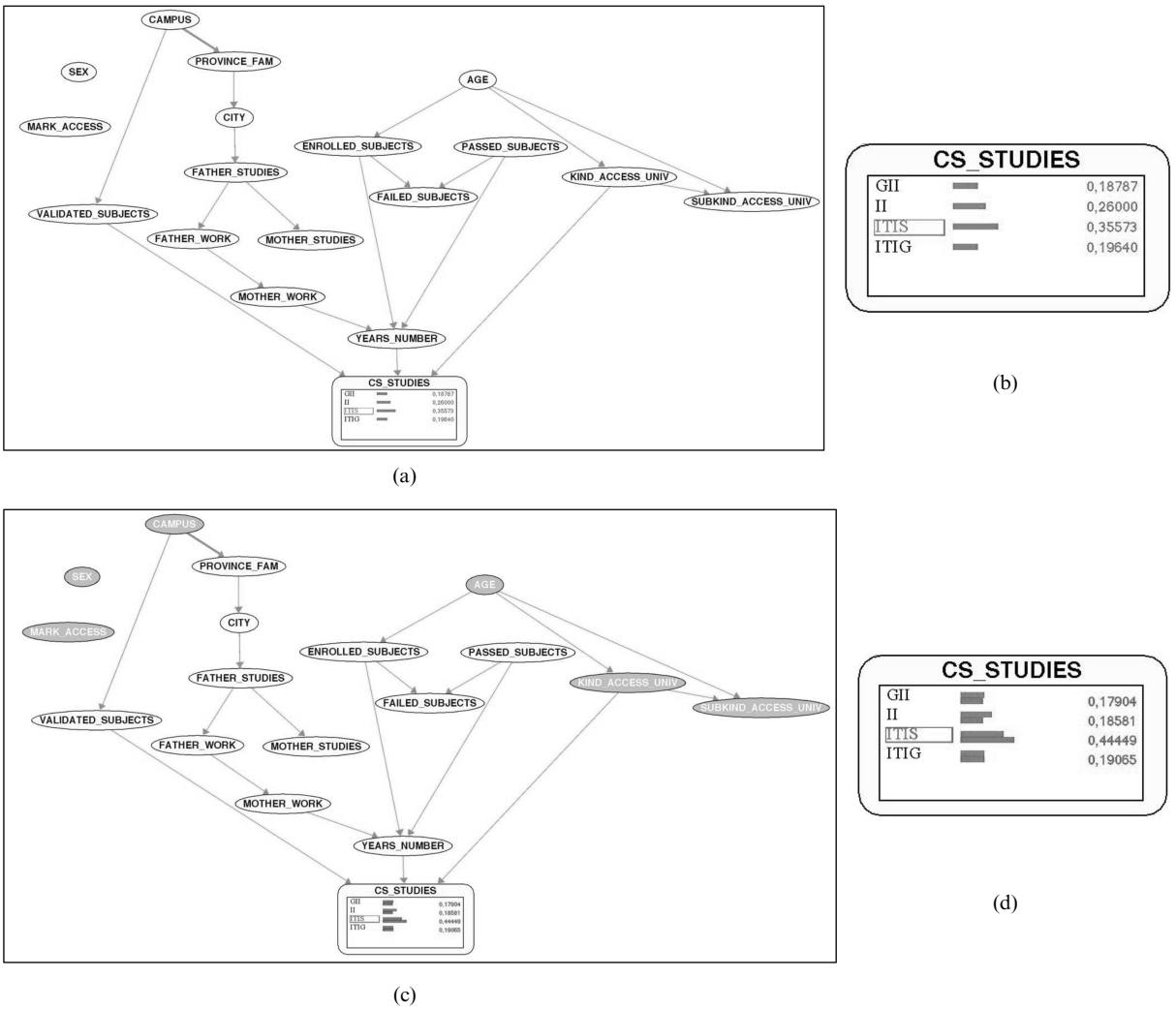

Fig. 7. (a) BN learned after applying K2 algorithm using Elvira software. (b) Zoomed variable *CS_STUDIES*. (c) BN After propagating evidence on variables *CAMPUS*, *SEX*, *AGE*, *KIND_ACESS_UNIV*, *SUBKIND_ACCESS_UNIV*. (d) Zoomed variable *CS_STUDIES* after evidence propagation.

5.2.2 Unfixed structure

In this stage, the K2 algorithm was applied to learn a BN from the database, obtaining the BN depicted in Fig. 7(a). It can be observed that variables *SEX* and *MARK_ACCESS* are independent from the rest which means that they do not provide any information to any variable. Moreover, the abandoned degree depends only on the number of validated subjects, the number of years enrolled and the kind of access to University. The links in this network should not be interpreted as causal relations, but as probabilistic dependencies, which can be easily interpreted with the help of Elvira. In this program, the type of dependence between the variables is exposed by the colour and the thickness of the links. The thickness of each link is proportional to the influence that each node transmits to another and the red (or darker) colour represent positive probabilistic dependencies, that is, as the parent takes greater values, the probability of the child taking greater values increases. For example, there is a positive dependency from *CAMPUS* to *VALIDATED_SUBJECTS*, which means that in Albacete the number of validated subject is greater than in Ciudad Real, since the values of *CAMPUS* are ordered alphabetically (Albacete, Ciudad Real).

Anyway, the same reasoning processes were performed in this network and, after propagating the same evidence of the previous examples (Figs. 5(c) and 6.c)), the distribution probability for C*S_STUDIES* variable is shown in Fig. 7(d). Besides, the total abduction process provided a probability of 0.00015 for the most probable profile whose configuration was like the one provided by the NB model except that the campus in which the student was enrolled and the province of family residence was Ciudad Real, his father's and mother's levels of employment were 1 and 4, respectively, the number of failed subjects was between 6 and 10, ant the CS studies that he left was ITIS.

Still, it was decided to use PC algorithm to learn a new model, shown in Fig. 8. Both the relations graph and the prior and posterior probabilities were different from the ones obtained with the algorithm K2. In this PC model, the variable *CS_STUDIES* depend only on the number of validated subjects. The most probable profile obtained had a probability of 0.00029 and it was very similar to the configuration provided by the K2 algorithm except that his city and province was Albacete, his age was greater than 30, his father's work was Level 5, the number of enrolled subjects was between 1 and 5 and the number of passed subjects was between 0 and 5.

6. Results and discussion

With the aim of making informed choices, it is recommended to develop several models and compare their properties both empirical and conceptually before deciding which one is the best that fits the stated objective [76, 93].

The accuracy of the models obtained in the previous section was compared by 5-fold cross validation (given the limited amount of cases) using the Elvira program. Table 1 shows the values obtained for the logarithm of the probability score which allows to conclude that K2 has better learning performance than the others, as it has also

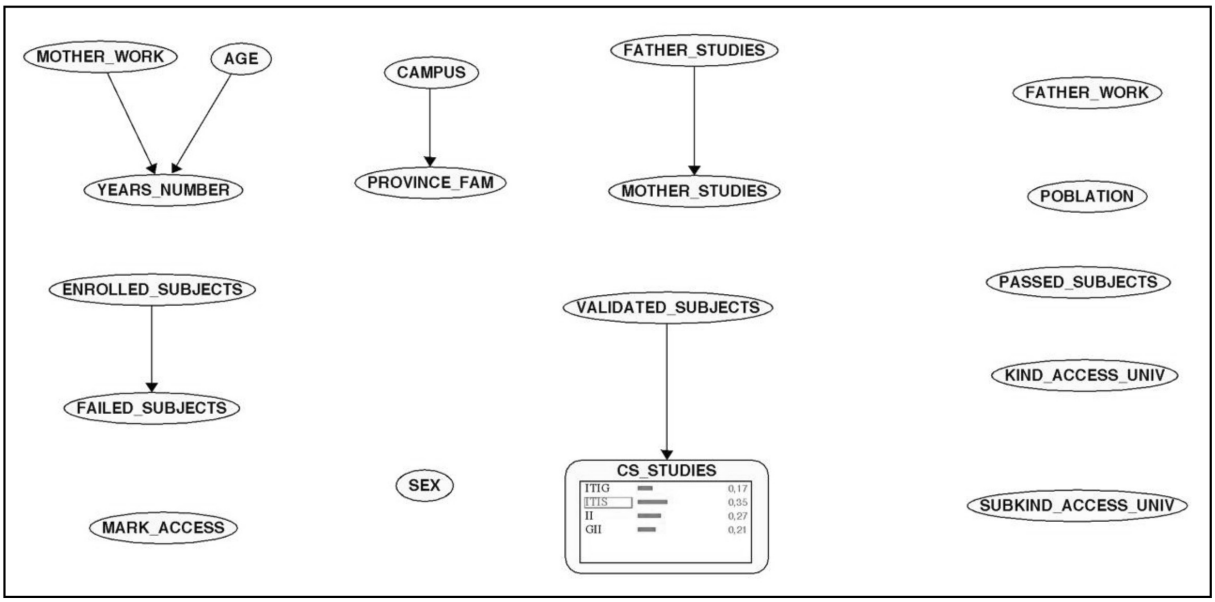

Fig. 8. BN learned after applying PC algorithm showing *CS_STUDIES* probabilities.

Table 1. Log-likelihood score with 5-fold cross validation

PC	K2	NB	TAN
–9.59	–8.71	–9.88	–10.54

Table 2. Number of links of the networks learned by each method

PC	K2	NB	TAN
6	18	17	33

been found in other works in the educative context [38, 76, 83].

The complexity of the learned structures can be measured by the number of links appearing in each graph, shown in Table 2. It can be seen that the most complex model is the one obtained by TAN algorithm and the simplest model is provided by PC algorithm. However, as Fig. 8 shows, the graph does not reflect every dependence among variables, undoubtedly because of the data base has not enough number of records to identify through PC algorithm all those relations. With respect to K2 and NB models, although both have a similar number of links, K2 graph is easier to understand and more intuitive because there are no crosses between edges.

Therefore, given the obtained results and taking into account that the objective of this research work is not only to get a good classification model, but also to identify the relationships between data in the data base that explain CS studies dropout, it can be concluded that the best model that explain is the Bayesian network learned through K2 algorithm (Fig. 7). However, the dependencies identified in the network learned through PC algorithm (Fig. 8) also appear in the network learned through TAN algorithm (Fig. 6) and K2 algorithm (Fig. 7). This fact suggests that those dependencies clearly underlie the data and can be interpreted as that the number of years that a student is enrolled at university before leaving it depends on his/her age and his/her mother's work; the specific CS degree left depends directly on the number of validated subjects; the number of failed subjects depends on the number of enrolled subjects; and the mother's studies depend on the father's studies.

Moreover, from K2 network some other interesting information can be extracted:

- *Gender* and *mark access* of students who leave CS Studies are not relevant factors affecting the left degree. This fact makes sense because in the case of sex, 81% of the students in the enrolment database and 85% in the abandonment database were men; regarding mark access, 82% of leaving students have the same mark, between 5 and 6.
- The specific *CS degree* left depends directly not only on the *number of validated subjects* but also on *the number of years* enrolled and the *kind of access* to university; being these factors dependent on the student *age*. Therefore, the *CS degree* left depends also on the student's *age*.
- The number of *failed subjects* depends not only on the number of *enrolled subjects* but and the number of *passed subjects*, which is completely logical.
- The *number of years* a student remains enrolled before dropping out his/her studies depends not only on his/her *mother's work* but also on the number of *enrolled subjects* and the number of *passed subjects*, what makes sense because these factors can be directly related to the economic situation of the student.
- The number of *validated subjects* depends on the *campus* enrolled, which may be due to the differences in the curricula of both campus.
- The *campus* where the student is enrolled affects the *province* of family residence, which is coherent because students tend to enrol at the campus closest to their family residence.

Regarding the profiles, Table 3 resumes the configurations obtained for the eighteen variables for each of the four earned models, represented in the rows. The columns are labelled with the abbreviated names of the variables and the last one contains the probability computed for all variables simultaneously.

The low probabilities for the profiles reveal a great heterogeneity of the data, mainly due to the high granularity of some variables, such as the province of family residence, the parental professions and their level of studies. However, the most probable configuration, with a probability of 0.00029, is the one provided by PC algorithm. This profile correspond to a male (M) aged between 31 and 40 years, enrolled in Albacete (AB) campus, whose family residence is in a town (T) in Albacete province, who accessed university by pre-registration (PRI) and Spanish selectivity (SEL) obtaining an access mark among 5 and 6 (5, 6); his parent

Table 3. Most probable configurations and their probabilities

	CS	*CP*	*S*	*A*	*PF*	*C*	*KAU*	*SAU*	*MA*	*FS*	*MS*	*FW*	*MW*	*#ES*	*#PS*	*#VS*	*#FS*	*YN*	P(C)
NB	GII	AB	M	20–25	AB	C	PR1	SEL	5–6	PRI	PRI	L7	UN	6–10	0–5	0–5	0–5	1	**0.00001**
TAN	GII	AB	M	20–25	AB	C	PR1	SEL	5–6	PRI	PRI	L2	L4	6–10	0–5	0–5	6–10	1	**0.00007**
K2	ITIS	CR	M	20–25	CR	T	PR1	SEL	5–6	PRI	PRI	L2	L5	6–10	0–5	0–5	6–10	7–13	**0.00015**
PC	ITIS	AB	M	31–40	AB	T	PR1	SEL	5–6	PRI	PRI	L2	L5	1–5	0–5	0–5	0–5	7–13	**0.00029**

studies are primary (PRI) and his father has a work level of 2 (L2) and his mother, a level 5 (L5); the number of enrolled was among 1 and 5 and the number of passed, validated and failed subjects was between 0 and 5; finally, the number of years enrolled in university before leaving his studies was between 7 and 13 and the CS degree left was ITIS.

The profile offered by K2 algorithm was very similar, although this differs in the student age, between 20 and 25; the enrolled campus and family residence, being CR, the number of enrolled and failed subjects, which are among 6 and 10; and the configuration probability, which is 0.00015. These two profiles are closer to reality than those provided by NB and TAN algorithms because 35% of students drop out ITIS degree compared to the 22% that leave GII. Also, 40% of students who leave CS studies are from Albacete and 39% are from Ciudad Real (CR).

7. Conclusions

In this paper, we have addressed the analysis of the student who leaves CS studies in the UCLM by using Bayesian networks. With such aim, different Bayesian network models have been developed from data obtained from the enrolments database of the University with the use of Elvira program, which allows the learning and subsequent editing of the network obtained and, also, provides facilities to interpret the results. These models have been obtained through the application of the most frequently used learning algorithms, which are, NB, TAN, PC and K2. The classification models have been compared by 5-fold cross validation and the best accuracy was obtained through the K2 network, i.e., the Bayesian network provided by K2 algorithm. Besides, its graph specifies the most informational dependencies among the variables.

On the other hand, considering both K2 and PC models, it can be deduced that gender and access mark are factors that do not affect the specific abandoned CS studies, but this depends directly on the number of validated subjects, the number of years that a student is enrolled at university and the kind of access to university; being these three features directly dependent on the age of student dropping out CS studies.

Regarding the profile of students that desert CS university studies, both PC and K2 networks offer similar configurations, being more probable and realistic than those obtained using the NB and TAN algorithms.

Finally, taking into account the power of Bayesian networks and the feasibility of the described methodology to develop these probabilistic models from a data base, authors are preparing a larger database, with many more records and more information of each student, to extract a model that allows to approach the problem of the abandonment of the students of Computer science from other different points of view.

Acknowledgments—This work was supported in part by the Spanish Ministry of Economy and Competitiveness under Grant TIN2015-66731-C2-2-R and in part by the Government of the Junta de Comunidades de Castilla-La Mancha under Grant PPII-2014-021-P. Authors would like to thank Dr. Serafín Moral for his invaluable help with learning algorithms and model validation with Elvira program.

References

1. L. Aulck, N. Velagapudi, J. Blumenstock and J. West, Predicting Student Dropout in Higher Education, *Proceedings of the ICML Workshop on #Data4Good: Machine Learning in Social Good Applications*, New York, 2016, pp. 16–20.
2. V. Tinto, Dropout from Higher Education: A Theoretical Synthesis of Recent Research, *Review of Educational Research*, **45**(1), 1975, pp. 89–125.
3. G. Kostopoulos, S. Kotsiantis and P. Pintelas, Estimating Student Dropout in Distance Higher Education Using Semi-Supervised Techniques, *Proceedings of the 19th Panhellenic Conference on Informatics*, 2015, pp. 38–43.
4. I. Witten, E. Frank, M. Hall and C. Pal, *Data Mining: Practical machine learning tools and techniques*, Morgan Kaufmann, 2016.
5. C. Romero and 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)*, **40**(6), 2010, pp. 601–618.
6. D. Kabakchieva, Predicting Student Performace by Using Data Mining Methods for Classification, *Cybernetics and Information Technologies*, **13**(1), 2013, pp. 61–72.
7. G. S. Abu-Oda and A. El-Halees, Data Mining in Higher Education: University Student Dropout Case Study, *International Journal of Data Mining & Knowledge Management Process*, **5**(1), 2015, pp. 15–27.
8. G. Dekker, M. Pechenizkiy and J. Vleeshouwers, Predicting students drop out: A case study, *Proceedings of the 2nd International Conference on Educational Data Mining*, 2009, pp. 41–50.
9. D. Dursun, Predicting student attrition with data mining methods, *Journal of College Student Retention: Research, Theory & Practice*, **13**(1), 2011, pp. 17–35.
10. V. Patil, S. Suryawanshi, M. Saner and V. Patil, Student performance prediction using classification data mining techniques, *International Journal for Research in Emerging Science and Technology*, **4**(4), 2017, pp. 15–18.
11. A. Bowers, Grades and Graduation: A Longitudinal Risk Perspective to Identify Student Dropouts, *The Journal of Educational Research*, **103**(3), 2010, pp. 191–207.
12. C. Burgos, M. L. Campanario, D. de la Peña, J. A. Lara, D. Lizcano and M. A. Martínez, Data mining for modeling students' performance: A tutoring action plan to prevent academic dropout, *Computers & Electrical Engineering*, **66**, 2018, pp. 541–556.
13. O. D. Oyerinde and P. A. Chia, Predicting Students' Academic Performances–A Learning Analytics Approach using Multiple Linear Regression, *International Journal of Computer Applications*, **157**(4), 2017, pp. 37–44.
14. M. Quadri and N. Kalyankar, Drop Out Feature of Student Data for Academic Performance using Decision Tree Techniques, *Global Journal of Compuer Science and Technology*, **10**(2), 2010, pp. 1–4.
15. S. Sivakumar, S. Venkataraman and R. Selvaraj, Predictive Modeling Student Dropout Indicators in Educational Data

Mining using Improved Decision Tree, *Indian Journal of Science and Technology*, **94**(4), 2016, pp. 1–5.
16. B. A. Pereira, A. Pai and C. Fernandes, A Comparative Analysis of Decision Tree Algorithms for Predicting Student's Performance, *International Journal of Engineering Science*, **7**(4), 2017, pp. 10489–10492.
17. R. R. Kabra and R. S. Bichkar, Performance prediction of engineering students using decision trees, *International Journal of Computer Applications*, **36**(11), 2011, pp. 8–12.
18. R. Asif, A. Merceron, S. A. Ali and N. G. Haider, Analyzing undergraduate students' performance using educational data mining, *Computers & Education*, **113**, 2017, pp. 177–194.
19. C. Marquez-Vera, A. Cano, C. Romero and S. Ventura, Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data, *Applied Intelligence*, **38**(3), 2013, pp. 315–330.
20. E. Yukselturk, S. Ozekes and Y. Türel, Predicting dropput student: an application of data mining methods in an online education program, *European Journal of Open, Distance and e-Learning*, **17**(1), 2014, pp. 118–133.
21. E. Costa, B. Fonseca, M. Almeida, F. Ferreira and J. Rego, Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses, *Computers in Human Behaviour*, **73**, 2017, pp. 247–256.
22. I. Lykourentzou, I. Giannoukos, V. Nikolopoulos, G. Mpardis and V. Loumos, Dropout prediction in e-learning courses through the combination of machine learning techniques, *Computers & Education*, **53**(3), 2009, pp. 950–965.
23. J. Fiaidhi, The next step for learning analytics, *IT Professional*, **16**(5), 2014, pp. 4–8.
24. P. Leitner, M. Khalil and M. Ebner, Learning Analytics in Higher Education—A Literature Review, In Peña-Ayala A. (eds) *Learning Analytics: Fundaments, Applications and Trends*, vol. 94, Springer, New York, 2017, pp. 1–23.
25. R. Baker and P. Inventado, Educational data mining and learning analytics, In J. Larusson and B. White (eds), *Learning Analytics. From Research to Practice*, Springer, New York, 2014, pp. 61–75.
26. D. Clow, An overview of learning analytics, *Teaching in Higher Education*, **18**(6), 2013, pp. 683–695.
27. Z. Papamitsiou and A. Economides, Learning Analytics and Educational Data Mining in Practice: A Systematic Literature Review of Empirical Evidence, *Educational Technology & Society*, **17**(4), 2014, pp. 49–64.
28. S.-F. Tseng, C.-Y. Chou, Z.-H. Chen and P.-Y. Chao, Learning Analytics: An Enabler for dropout Prediction, *Proceedings of the 22nd International Conference on Computers in Education*, Japan, 2014, pp. 286–288.
29. M. Morales and A. Salmeron, Análisis del alumnado de la Universidad de Almería mediante redes bayesianas, *Actas del 27 Congreso Nacional de Estadística e Investigación Operativa*, Lleida, 2003.
30. A. Fernández, M. Morales, C. Rodríguez and A. Salmerón, A system for relevance analysis of performance indicators in higher education using Bayesian networks, *Knowledge and Information Systems*, **27**(3), 2011, pp. 327–344.
31. L. Di Pietro, R. G. Mugion, F. Musella, M. F. Renzi and P. Vicard, Reconciling internal and external performance in a holistic approach: A Bayesian network model in higher education, *Expert Systems with Applications*, **42**, 2015, pp. 2691–2702.
32. J. C. Corey, *Bayesian networks with expert elicitation as applicable to student retention in institutional research*, PhD Dissertation, Georgia State University, 2016.
33. P. Naïm, P. Wuillemin, P. Leray, O. Pourret and A. Becker, *Réseaux bayésiens*, 3rd Edition, Eyrolles, 2007.
34. O. Pourret, P. Naïm and B. Marcot, *Bayesian Networks: A practical guide to applications*, Wiley, 2008.
35. E. Castillo, J. Gutierrez and A. Hadi, *Expert Systems and Probabilistic Networks Models*, Springer Verlag, New York, 1997.
36. J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kauffman, San Mateo, 1988.
37. E. Millan, T. Loboda and J. Perez-de-la-Cruz, Bayesian networks for student model engineering, *Computers & Education*, **55**(4), 2010, pp. 1663–1683.
38. A. Fernandez, M. Morales, C. Rodriguez and A. Salmeron, A system for relevance analysis of performance indicators in higher education using Bayesian networks, *Knowledge and Information Systems*, **27**, 2011, pp. 327–344.
39. F. Jensen, *Bayesian networks and decision graphs*, Springer, New York, 2001.
40. S. Lauritzen and D. Spiegelhalter, Local computations with probabilities on graphical structures and their application to expert systems, *Journal of the Royal Statistical Society, Series B*, **50**, 1988, pp. 157–224.
41. A. Cano, S. Moral and A. Salmeron, Penniless propagation in join trees, *International Journal of Intelligent Systems*, **15**, 2000, pp. 1027–1059.
42. G. Shachter and P. Shenoy, Probability propagation, *Annals of Mathematical and Artificial Intelligence*, **2**(1–4), 1990, pp. 327–351.
43. G. Cooper, The computational complexity of probabilistic inference using Bayesian belief networks, *Artificial Intelligence*, **42**(2–3), 1990, pp. 393–405.
44. R. Bouckaert, E. Castillo and J. Gutiérrez, A modified simulation scheme for inference in Bayesian networks, *International Journal of Approximate Reasoning*, **14**(1), 1996, pp. 55–80.
45. R. Shachter and M. Peot, Simulation approaches to general probabilistic inference on belief networks, *Proceedings of the Fifth Annual Conference on Uncertainty in Artificial Intelligence*, North Holland, Amsterdam, 1990, pp. 311–318
46. D. Nilsson, An efficient algorithm for finding the *M* most probable configurations in probabilistic expert systems, *Statistics and Computing*, **8**, 1998, pp. 159–173.
47. C. Lacave and F. Díez, Knowledge acquisition in PROSTANET, a Bayesian network for diagnosing prostate cancer, In V. Palade, R.J. Howlett and L. Jain (eds), *Knowledge-Based Intelligent Information and Engineering Systems, Lecture Notes in Computer Science*, vol. 2774, Springer, Berlin, 2003, pp. 1345–1350.
48. R. Neapolitan, *Learning Bayesian Networks*, Pearson, 2004.
49. D. Heckerman, D. Geiger and D. Chickering, Learning Bayesian networks: The combination of knowledge and statistical data, *Machine Learning*, **20**(3), 1995, pp. 197–243.
50. D. Spiegelhalter, R. Frankling and K. Bull, Assessment, criticism and improvement of imprecise subjective probabilities, *Proceedings of the Fifth Annual Conference on Uncertainty in Artificial Intelligence*, North Holland, Amsterdam, 1990, pp. 285–294
51. A. O'Hagan, C. Buck, A. Daneshkhah, J. Eiser, P. Garthwaite, D. Jenkinson, J. Oakley and T. Rakow, *Uncertain Judgements: Eliciting Expert's Probabilities*, Wiley, 2006.
52. I. Bermejo, J. Oliva, F. Díez and M. Arias, Interactive learning of Bayesian networks using OpenMarkov, *Proceedings of the Sixth European Workshop on Probabilistic Graphical Models*, Granada, Spain, 2012, pp. 27–34.
53. OpenMarkov, http://www.openmarkov.org, Accessed 4 January 2018.
54. M. Minski, Steps toward artificial intelligence, In E.A. Feigenbaum and J. Feldman (eds), *Computers and Thoughts*, MIT Press, Cambridge, MA, USA, 1995, pp. 406–450.
55. N. Friedman, D. Geiger and M. Goldszmit, Bayesian networks classifiers, *Machine Learning*, **29**, 1997, pp. 131–163.
56. G. Cooper and E. Herskovits, A Bayesian method for the induction of probabilistic networks from data, *Machine Learning*, **9**(4), 1992, pp. 309–347.
57. P. Spirtes, C. Glymour and R. Scheines, *Causation, Prediction and Search*, Springer, New York, 1993.
58. R. Jin, Y. Breitbart and C. Muoh, Data discretization unification, *Knowledge and Information Systems*, **19**(1), 2009, pp. 1–29.
59. A. Dempster, N. Laird and D. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society. Series B*, **39**(1), 1977, pp. 1–38.
60. A. Sharabiani, F. Karim, A. Sharabiani, M. Atanasov and H. Darabi, An enhanced Bayesian network model for prediction of students' academic performance in engineering

programs, *Proceedings of the IEEE Global Engineering Education Conference*, Istanbul, 2014, pp. 832–837.
61. R. Bekele and W. Menzel, A Bayesian approach to predict performance of a student (BAPPS): A case with ethiopian students, de *Proceedings of the International Conference on Artificial Intelligence and Applications*, Innsbruck, Austria, 2005.
62. Elvira Consortium, Elvira: an environment for probabilistic graphical models, *Proceedings of the First European Workshop on Probabilistic Graphical Models*, Cuenca, Spain, 2002, pp. 222–230.
63. Elvira System, http://leo.ugr.es/elvira, Accessed 4 January 2018.
64. S. Andersen, K. G. Olesen and F. Jensen, HUGIN— a shell for building Bayesian belief universes for expert systems, *Readings in uncertain reasoning*, Morgan Kauffman, San Francisco, USA, 1990, pp. 332–337.
65. Hugin Expert, https://www.hugin.com, Accessed 4 January 2018.
66. K. B. Korb and A. E. Nicholson, *Bayesian Artificial Intelligence*, Second ed. London: Chapman & Hall, 2010.
67. C. A. Pollino, O. Woodberry, A. E. Nicholson, K. B. Korb and B. T. Hart, Parameterisation and evaluation of a Bayesian network for use in an ecological risk assessment, *Environmental Modelling & Software*, **22**(8), 2007, pp. 1140–1152.
68. R. E. Kass and L. Wasserman, The selection of prior distributions by formal rules, *Journal of the American Statistical Association*, **91**(435), 1996, pp. 1343–1370.
69. L. Lalande, L. Bourguignon, C. Carlier and M. Ducher, Bayesian networks: A new method for the modeling of bibliographic knowledge, *Medical and Biological Engineering & Computing*, **51**(6), 2013, pp. 269–293.
70. B. Marcot, Metrics for evaluating performance and uncertainty of Bayesian network models, *Ecological Modelling*, **230**(10), 2012, pp. 50–62.
71. S. Geisser, The predictive sample reuse method with applications, *Journal of the American Statistical Association*, **70**(350), 1975, pp. 320–328.
72. T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, New York, 2009.
73. S. Geman, E. Bienenstock and R. Doursat, Neural networks and the bias/variance dilemma, *Neural Computation*, **4**(1), 1992, pp. 1–58.
74. W. Xing, R. Guo, E. Petakovic and S. Goggins, Participation-based student final performance prediction model through interpretable Genetic Programming: Integrating learning analytics, educational data mining and theory, *Computers in Human Behavior*, **47**, 2015, pp. 168–181.
75. C. Lacave, M. Luque and F. Díez, Explanation of Bayesian networks and influence diagrams in Elvira, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **37**(4), 2007, pp. 952–965.
76. B. W. Oviedo Bayas, *Modelos gráficos probabilísticos aplicados a la predicción del rendimiento en educación*, PhD Dissertation, Granada, Spain, 2016.
77. B. Bakhshinategh, O. R. Zaiane, S. ElAtia and D. Ipperciel, Educational data mining applications and tasks: A survey of the last 10 years, *Education and Information Technologies*, **22**, 2017, pp. 1–17.
78. A. Daud, N. R. Aljohani, R. A. Abbasi, M. D. Lytras, F. Abbas and J. S. Alowibdi, Predicting student performance using advanced learning analytics, *Proceedings of the 26th International Conference on World Wide Web Companion*, Republic and Canton of Geneva, Switzerland, 2017, pp. 415–421.
79. A. Dutt, M. A. Ismail and T. Herawan, A Systematic Review on Educational Data Mining, *IEEE Access*, **5**, 2017, pp. 15991–16005.
80. L. Barbosa, S. Serra and G. Zimbrão, Towards automatic prediction of student performance in STEM undergraduate degree programs, *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, 2015, pp. 247–253.
81. M. J. Culbertson, Bayesian networks in educational assessment: The state of the field, *Applied Psychological Measurement*, **40**(1), 2016, pp. 3–21.
82. A. Peña-Ayala, Educational data mining: A survey and a data mining-based analysis of recent works, *Expert systems with applications*, **41**(4), 2014, pp. 1432–1462.
83. B. Oviedo, S. Moral and A. Puris, A hierarchical clustering method: Applications to educational data, *Intelligent Data Analysis*, **20**(4), 2016, pp. 933–951.
84. C. Conati, A. Gertner e K. VanLehn, Using Bayesian networks to manage uncertainty, *Journal of user modeling and user-adapted interaction*, **12**(4), 2002, pp. 371–417.
85. J. C. Dunn, Bayesian Networks with Expert Elicitation as Applicable to Student Retention in Institutional Research, PhD Dissertation, Georgia State University, Atlanta, USA, 2016.
86. S. B. Kotsiantis, C. J. Pierrakeas and P. E. Pintelas, Preventing student dropout in distance learning using machine learning techniques, In V. Palade, R.J. howlett and L. Jain (eds), *Knowledge-Based Intelligent information and Engineering Systems. Lecture Notes in Computer Science*, **2774**, Berlin, 2003, pp. 267–274.
87. E. Er, Identifying at-risk students using machine learning yechniques: A case study with IS 100, *International Journal of Machine Learning and Computing*, **2**(4), 2012, pp. 476–480.
88. S. Pal, Mining educational data using classification to decrease dropout rate of students, *International Journal of Multidisciplinary Sciences and Engineering*, **3**(5), 2012, pp. 35–39.
89. E. Osmanbegović and M. Suljic, Data mining approach for predicting student performance, *Journal of Economics and Business*, **10**(1), 2012, pp. 3–12.
90. S. Taruna and M. Pandey, An empirical analysis of classification techniques for predicting academic performance, *Proceedings of the 2014 IEEE International Advance Computing Conference*, 2014, pp. 523–528.
91. R. Bukralia, A. Deokar, S. Sarnikar and M. Hawkes, Using machine learning techniques in student dropout prediction, In H. Burley (eds), *Cases on Institutional Research Systems*, IGI Global, USA, 2012, pp. 117–131.
92. C. Chiok, Predicción del rendimiento académico aplicando técnicas de minería de datos, *Anales Científicos*, **78**(1), 2017, pp. 26–33.
93. V. Nikolovski, R. Stojanov, I. Mishkovski, I. Chorbev and G. Madjarov, Educational data mining: Case study for predicting student dropout in higher education, *Proceedings of the 12th International Conference on Informatics and Information Technologies*, 2015.

Carmen Lacave has a degree in CC Mathematics (1990) by the University Complutense of Madrid (Spain) and a PhD in Sciences (2003) by the UNED (Spain). She is currently a full professor in the area of Computer Languages and Systems at the University of Castilla-La Mancha. Belonging to the research group CHICO (Computer-Human Interaction and Collaboration) of the University of Castilla-La Mancha, her work focuses on the application of Information Technology and Artificial Intelligence to education.

Ana I. Molina has a degree in Computer Science (2002) and a PhD (2007) from the University of Castilla-La Mancha (Spain). Since 2003 she has been a member of the CHICO research group of the University of Castilla-La Mancha. In addition to teaching, her main areas of interest are Information Technologies applied to education, the design and specification of collaborative interfaces and the evaluation of educational resources through eye tracking techniques.