Modeling Student Performance in Higher Education Using Data Mining

Chapter · November 2014

DOI: 10.1007/978-3-319-02738-8_4

2 authors:

Huseyin Guruler
Mugla Üniversitesi
28 PUBLICATIONS **109** CITATIONS
SEE PROFILE

Ayhan Istanbullu
Balikesir University
23 PUBLICATIONS **165** CITATIONS
SEE PROFILE

Chapter 4
Modeling Student Performance in Higher Education Using Data Mining

Huseyin Guruler and Ayhan Istanbullu

Abstract Identifying students' behavior in university is a great concern to the higher education managements (Kumar and Uma, Eur J Sci Res 34(4):526–534). This chapter proposes a new educational technology system for use in Knowledge Discovery Processes (KDP). We introduce the educational data mining (EDM) software and present the outcome of a test on university data to explore the factors having an impact on the success of the students based on student profiling. In our software system all the tasks involved in the KDP are realized together. The advantage of this approach is to have access to all the functionalities of the Structured Query Language (SQL) Server and the Analysis Services through a single developed software item, which is specific to the needs of a higher education institution. This model (Guruler et al., Comput Educ 55(1):247–254) aims to help educational organizations to better understand the KDPs, and provides a roadmap to follow while executing whole knowledge projects, which are non-trivial, involve multiple stages, possibly several iterations.

Keywords Educational data mining · Educational technology system and architectures · Student relationship management · Knowledge discovery software · Decision tree

H. Guruler (✉)
Department of Information Systems Engineering, Technology Faculty, Mugla Sitki Kocman University, 48000 Kötekli, Mugla, Turkey
e-mail: hguruler@mu.edu.tr

A. Istanbullu
Department of Computer Engineering, Engineering and Architecture Faculty, Balikesir University, 10145 Cagiş, Balikesir, Turkey
e-mail: iayhan@balikesir.edu.tr

A. Peña-Ayala (ed.), *Educational Data Mining*,
Studies in Computational Intelligence 524, DOI: 10.1007/978-3-319-02738-8_4,

Abbereviations

CM	Correlation matrices
DBMS	Database management system
DM	Data mining
DT	Decision tree
DTS	Data transformation services
EDM	Educational data mining
GPA	Grade point average
KDD	Knowledge discovery in databases
KDP	Knowledge discovery process
MDAC	Microsoft data access components
MDT	Microsoft decision tree
OLAP	On-line analytical processing
PDCA	Plan-do-check-act
SKDS	Student knowledge discovery software
SRM	Student relationship management
SQL	Structured query language

4.1 Introduction

Appropriate decisions can be made by effectively analyzing and managing the growing volume of data. Gaining information from business data started with data collection in the 1960s; this type of data collection answered questions related to the past. In the 1980s, with the development of relational databases, data access methods were introduced. In the 1990s, data warehousing and decision support systems were created based on multi-dimensional databases and On-line Analytical Processing (OLAP). Today, data mining (DM) produces a particular enumeration of patterns in data. This should be understandable and usable by the business end user. To accomplish this, there is a typical data-driven business process consisting of multiple stages between multiple servers and data extracts, preprocessing, and conversions with advanced algorithms, multi-processor computers and massive databases [1].

DM is a new data-oriented technology, which is able to discover valuable interactions in human activities using computer implementations. For this purpose, an automated-process to uncover trends, patterns, and relationships from accumulated electronic traces is used to collect the data [2].

Recently, knowledge discovery in databases (KDD) methodologies have been used to enhance and evaluate higher education tasks [3]. This process, contributes to the enhancement of the quality of a higher educational system by evaluating student data. Analyzing and manipulating the existing data with respect to

predefined goals provide high quality, student-specific, and student-centered education for higher education institutions. Thus, DM promises better ways to produce higher quality in education, and greater satisfaction for student [4]. Moreover, Web-based systems routinely collect vast quantities of data on user patterns, and DM methods can be applied to these databases. Newly developed web-based educational technologies, also offer researchers unique opportunities to evaluate the factors affecting students' learning capacity which is an important element of their academic success [5].

Another fundamental role of universities is to raise the quality of education, as well as producing and disseminating information. Recorded data in universities contains valuable information regarding students, which is usually used for official procedures such as producing transcripts. In fact, this data could also be used in academic guidance of students using a separate discovery investigation to extract information relevant to the individual student's progress [6]. Additionally, competitive advantages could be obtained by identification of the students' demands through the available data. In this direction, some models have been proposed and implemented. One of them demonstrates how DM can be utilized in a higher educational system to improve the efficiency and effectiveness of the traditional processes [7]. The other model was combined with a deterministic model to analyze the students' results over the 2 or 3 semesters in the academic year in a private educational institution [8].

In the increasing commercialized education environment, higher educational institutions need to become more efficient, provide a better quality service to deliver exceptional student experience [9]. Moreover students and their parents want an education that is tailored to their needs. Student Relationship Management (SRM) is one of the responses to these demands [10]. SRM can be described as a proactive management system which creates a single, holistic view of each student by bringing together different elements of data from various sources such as; academic departments, student services and independent systems such as finance and accommodation [11]. SRM is valuable when data is scattered across an institution, in different departments, in various file formats. It is designed to impact on every connection in the student lifecycle and integrates with an institutions current projects and systems, avoiding duplication and ensuring a fluid, step-change in student management.

This smarter student management uses predictive analytics that considers the mix of very different metrics on students and from this data can be confidently predicted their potential failure or success. The results can trigger action to bring proactive support to the learner and help remove the factors that lead to failure. Integrated profiles, analytics and tools to increase the quantity and quality of admissions across the institution, furthermore, the success rate of the establishment increases [11].

There are several approaches to KDP. A chapter in the book [12] describes the KDP, presents models, and explains why and how these could be used for a successful DM project. In the context of DM, Crisp-DM model is considered to be a significant standard, however, it is highly recommended in a technical project

report [13] that following a structured plan-do-check-act (PDCA) cycle in DM applications to achieve an optimized quality and success. The PDCA cycle as an approach to change and problem solving is very much at the heart of Deming's quality-driven philosophy [14]. The four phases in the PDCA Cycle are:

- Plan: Identifying and analyzing the problem.
- Do: Developing and testing a potential solution.
- Check: Measuring how effective the test solution was, and analyzing whether it could be improved in any way.
- Act: Implementing the improved solution fully.

Crisp-DM can be combined with the PDCA cycle in this study as presented in Fig. 4.1 in which, the framework of the PDCA cycle has eight stages, which captures all the facets of the DM tasks [15]. The major stages are: problem identification; gathering and selection of data; data preprocessing for missing, duplicate or erroneous information; selection of appropriate learning algorithms; preparation and processing of data; construction and evaluation of the models; interpretation of the discovered knowledge; and finally, taking action.

The cycle starts with the plan stage where a precise business objective and the related business problem or opportunity, is defined and the application domain is demarcated. This stage is very important since it determines the scope of the project. Then comes the do stage that includes the stages of the KDP which tends to be highly iterative and interactive [12].

The target data set is selected from a large database. After selecting the target data set, cleaning, preprocessing and reduction create the appropriate data set for further transformation and combination. After choosing the functions of DM and the DM algorithm(s) follow next the DM process is initiated. The outcome of this whole process is the discovered knowledge that is interpreted and evaluated for the business client.

This knowledge consists of the relationships and patterns found in the data which becomes the input for the check stage where the analyses are completed to assess whether the knowledge is applicable to the scope of the project. If the results are interesting and satisfactory, last stage is to act on the results by implementing the solution suggested by the KDD results. This cycle is continuous, as new or related problems arise over time the cycle becomes continuous and each time the mechanism will try to find solutions with the help of DM tools [16].

This chapter proposes a new EDM system 'Student Knowledge Discovery Software (SKDS)', introduces its architecture for use in KDPs and presents the outcome of a test on university data to explore the factors having an impact on the success of the students based on student profiling [17, 18]. SKDS is a specific system that integrates the EDM process with the database management system (DBMS) [19]. Although there are different commercial software applications available that are adapted to DM [20], our approach has some major advantages for educational institutions. First, the data analysis realizes where it is generated therefore; the analysis can be easily repeated on new inserted data. Secondly, the software can use the functions of the SQL Server and the Analysis Services, which

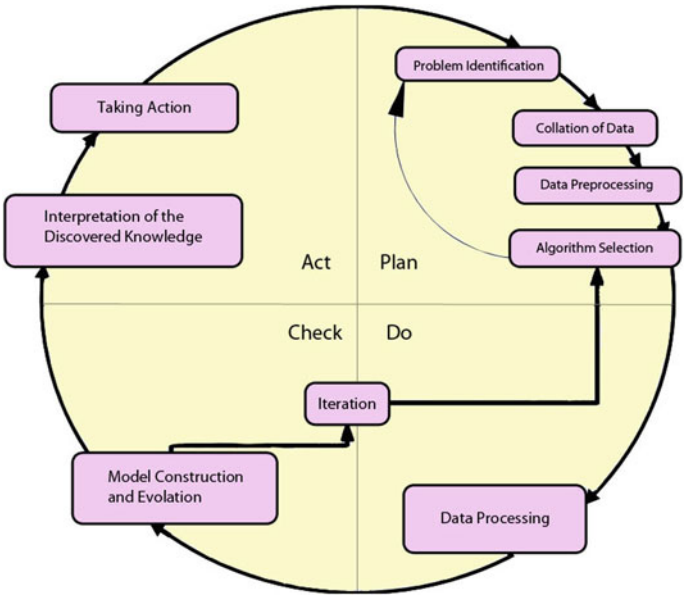

Fig. 4.1 The framework of PDCA cycle for DM

have programmed DM techniques. Thus, the user can perform tasks without any need for complicated Structured Query Language (SQL) statements. Finally, the specifically designed user interface makes it possible to follow the EDM process and to perform the database management activities in an easy and orderly manner. The end-user of this software must have fundamental skills for computer and database operations with the knowledge of EDM. The results can be assessed by the decision-makers, such as advisors and administrators.

The rest of the chapter is organized as follows: Sect. 4.2 presents the basic concepts of decision tree (DT) classification analysis. Section 4.3 introduces the system overview, user interface and architecture of SKDS. Section 4.4 gives notations related to our performed study. Section 4.5 presents the results of our EDM model. In Sect. 4.6 the conclusion enumerates the advantages and validates the proposed approach.

4.2 Background

There are basically two types of DM. The user creates an explicit or implicit hypothesis about the data in the "verification-driven" DM. Limited by the hypothesis, a query concerning the data is conducted and the results of this query are examined. If the result is positive, the process ends otherwise, a new query is formulated and the process iterates until the resulting data either verifies the

hypothesis or the user decides that the data is not valid for this approach [21]. Thus, little new information is created. Whereas, a well-designed DM tool is able to build the exploration of the data and to yield as many useful facts about the data as possible in the shortest amount of time.

The system searches data to find frequently occurring patterns, detect trends or produce generalizations about the data. The discovery is completed with very little (or no) guidance from the user and this uncovering of the facts is not a consequence of a haphazard event.

Discovery-driven DM is further divided into two categories. Descriptive (undirected) models provide information to gain an increased understanding of what is happening inside the data without a predetermined idea. The program takes the initiative to find interesting patterns in large databases, since there are so many patterns that the user would not be able to form the appropriate questions to ask. The power and usefulness of discovered results from the richness and quality of the discovered information. In an undirected DM, no variable is singled out as the target, the goal is to establish a relationship among all the variables such as clustering algorithms [22].

Classification is the task of examining the features of a newly presented object and assigning it to one of a predefined set of classes [23]. In this study, the decision tree (DT) classification was chosen since this structure offers the ability to easily generate rules, provide understandable models, and achieve a high level of integration with information technology processes because it requires little preprocessing of data [24].

4.2.1 The Decision Tree Classification Model

The DT classification is a supervised learning method that constructs a tree from a set of examples. It creates classification models by examining already classified data from a historical database and inductively finding a predictive pattern. This pattern can be used both to understand the existing data and predict how new instances will behave. It is a predictive model viewed as a tree consisting of decision nodes, branches and leaves. A decision node specifies a test to be carried out, which branches are to be supplied without losing any data.

The split decision is made at the node "in the moment", it is never revisited and also univariate. In addition, all splits are made sequentially, so each split is dependent on its predecessor. Thus, all future splits are dependent on the first split, which means the final solution could be very dissimilar if a different first split had been made. Each branch of the tree is a possible answer to the classification question and will lead either to another decision node or to the bottom of the tree, called a leaf node. The leaves are the partitions of the data set with their classification. The DT process starts at the root node and moves to each subsequent node until a leaf node is reached [25].

From a business perspective, DTs can be viewed as creating a segmentation of the original data set to predict some important piece of information (each segment would be one of the leaves of the tree). The predictive segments are similar with respect to the information being predicted and contain a description of the characteristics that define the predictive segment. Thus, although the DTs and algorithms may be complex, the results are easy-to-understand [26].

There are several major advantages as well as disadvantages in using DTs. The most important advantage of DTs are generating understandable roles no matter how complicated the inputs are. It is generally easy to follow any one path through the tree, so explaining the decisions along the way is also easy.

The computation cost for each split is minimal. In practice, algorithms tend to produce DTs with a low branching factor with simple tests at each node, so the tree does not grow too large and these tests translate into simple boolean and integer operations that are fast and inexpensive. Using DTs, the field, which is the best at splitting the training records, can be singled out for analysis. This will enable the user to determine which variable mostly influences their data. However, when there are a large number of factors affecting data it might be very difficult to determine specific factors; therefore DTs are not suited for numbers covering large intervals [27].

4.2.2 The Decision Tree Mechanism

DTs are built using recursive partitioning which is an iterative process of splitting the data up into partitions. Initially, the algorithm seeks to create a tree that works as perfectly as possible on all the available data, but this does not usually work. The process starts with a training set consisting of pre-classified records. In order to build a tree that distinguishes the classes, the best possible question to ask at each branch point of the tree has to be found. The goal is for the leaves of the tree to be as homogeneous as possible with respect to the prediction value. The diversity measure is calculated for the two partitions, and the best split is that with the largest decrease in diversity. After the tree has been grown to certain size, the algorithm has to check if the model overfits the data which it does by a cross validation approach. The tree size can be controlled via stopping rules limiting growth [28].

The quality of a tree depends on both its size and the classification accuracy [29]. The method first chooses a subset of the training examples to form a DT. If the tree does not give the correct answer for all the objects, a selection of the exceptions is added to the window and the process continues until the correct decision set is found.

The eventual outcome is a tree in which each leaf carries a class name, and each interior node specifies an attribute with a branch corresponding to each possible value of that attribute. Entropy is a measure commonly used in information theory. The higher the entropy of an attribute, the more uncertainty there is with respect to its outcomes. Thus, we would want to select attributes in order of increasing entropy, where the root node of our tree would correspond to the attribute with the

lowest entropy value. More information about the methodology and related measures of DTs can be found in [30, 31].

4.3 System Overview, Software Interface and Architecture

In this EDM application, the Microsoft Windows Server and the Microsoft SQL Server were used as an operating system and a relational DBMS, respectively. In addition, the Analysis Services in the SQL Server were used to create and validate DM models, the Microsoft Data Access Components (MDAC) were used to access the data, Angoss DM consumer controls were used to display the models and validation results. Furthermore, the SKDS was developed using the programming technologies (plugins, controls and tools) of the Microsoft Visual Basic.

While implementing some tasks related to the EDM process, the SQL Server tools were called upon for example, in the transformation stage where the Data Transformation Services (DTS) import/export wizard accesses the data set and then transforms the column values. Moreover, the DTS creates predictions based on the DM model and performs actions according to the results.

In the solution development phase of the study, a task sharing mechanism between the SQL Server and the Analysis Services has been developed in order to implement the tasks involved in every individual EDM stage. In fact, each of the tasks can be separately implemented either on the SQL server or the Analysis Services. In order to perform all the tasks together, SKDS was developed. SKDS was specifically designed for use with student demographic data. The purpose of SKDS is:

- To set an example for the EDM process.
- To access all the tasks from a single program.
- To access all functions of the SQL Server and the Analysis Services by means of programming techniques. In this way, the user can perform tasks without any need for complicated SQL statements.

The SKDS user interface is shown in Fig. 4.2. SKDS consists of three main sections; the database connection, data preparation and model development. In each section a button represents a task and takes the user either to a form or to a wizard related to this task. Since it is important to undertake the tasks in order of precedence, the interface was designed to indicate this order. Forms accessed through the user interface make it possible to follow the KDP from the perspective of the PDCA cycle and to perform the database management activities in an easy and orderly manner.

Figure 4.3 presents the SKDS working principle as a block diagram. This shows that the user (computer science professional familiar with the principles of knowledge discovery and student data that is to be used) who will perform the DM can access the forms directly (table management, sampling, cleaning, research and exploring, splitting, modeling, control and validating) or indirectly (transformation) for eight different tasks.

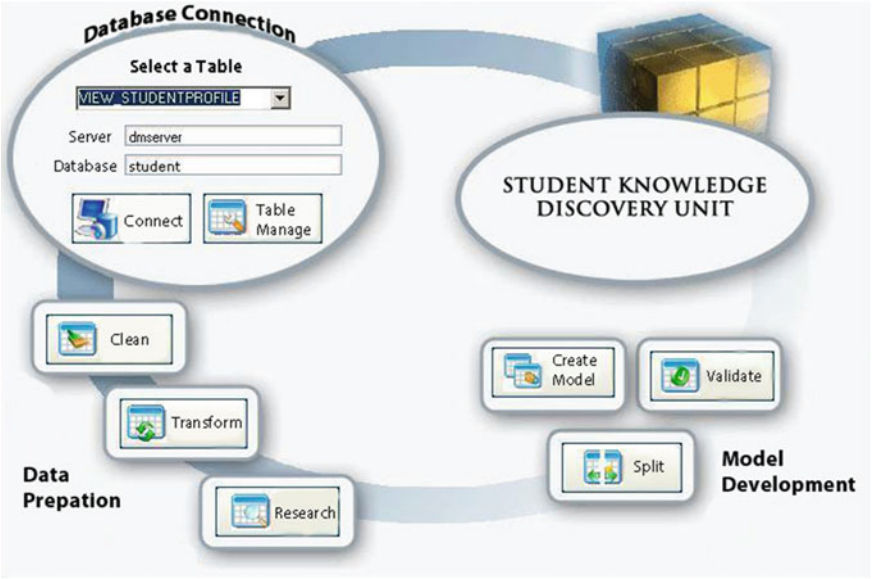

Fig. 4.2 SKDS user interfaces

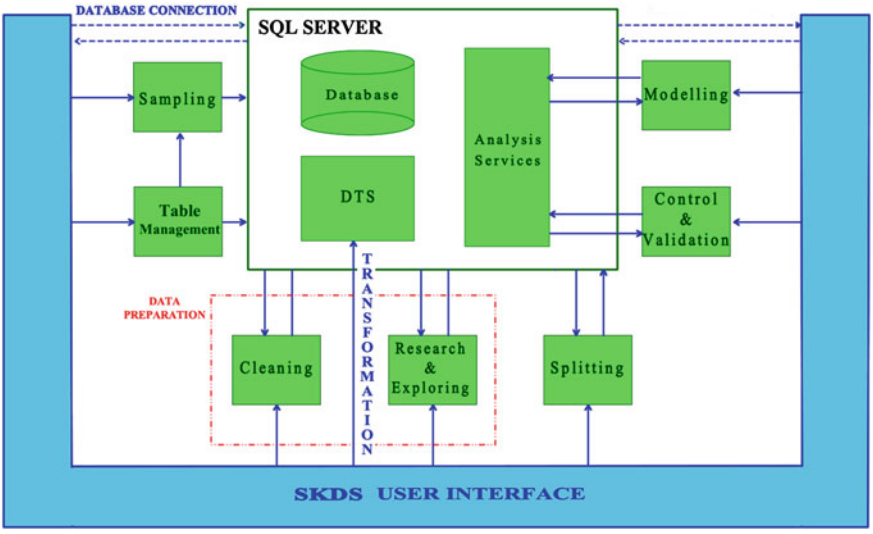

Fig. 4.3 SKDS architecture

In SKDS, first a database connection is made to the data set that contains the data on which an investigation is to be carried out. After the connection is established, the various editing and backup activities listed below can be

performed through table management form or sampling form. Depending on the data mining task the table management techniques are:

- Select specific columns to include in a new table (i.e. model table, test table).
- Drop tables from the data set that are no longer useful. This is part of the cleaning.
- Create a copy of an existing table but containing fewer rows. This is used to reduce the amount of time spent on the testing to find the best model.

The next stages are the cleaning, transformation, and data analyzing (research and exploring) tasks that constitutes the preparation part of the data. In the cleaning process, the following three automated tasks are performed after selection of some specifications mentioned below.

- For the columns containing a large number of null values; in order to decide which columns to include or exclude in the modeling, the percentage of null values for each column is calculated. The columns having more than the specified percentage of null value are excluded. Since the percentage value is empirically obtained to optimize the usability of input columns, the value can be easily changed.
- For the columns that have one (i.e. same content), a few (distinct values in a group with same content or a spread of neutral status) or too many (close to the number of records such as; students' address and phone numbers) distinct values; mean, minimum and maximum value and the number of distinct values for each column are calculated. For instance, if the mean, minimum and maximum values are equal, the column will have the same value for each record. This information is used to exclude the columns which appear not to be useful.
- For the outlying cell values (generally incorrectly entered data, such as income defined as very low or very high) that are not compatible with normal distribution are determined and the rows containing these values are highlighted. Then, according to the chosen solution, the rows can be excluded from the analysis or the related values can be replaced by the average value of those columns.

If the input columns have too many distinct values (categories), it is difficult to discover how this affects the target column. To overcome this problem discretization can be undertaken; this is a process of putting values into buckets so that there are a limited number of possible states. The buckets themselves are treated as ordered and discrete values. Both numeric and string columns can be discretized in the input columns using transformations. The DTS import/export assists the SQL Server, which carries out this transformation in two ways:

- The input columns having a large number of possibilities for the values are categorized exhaustively into a limited number of categories such as category 1, category 2, …, category k.
- In some cases in which the variables that may have some effect on target column are not in a single column or not directly available the transformations on these input columns can be used to create new useful columns.

The last procedure of data preparation is the data analyzing, before splitting the data, consists of the research and explore. On these tasks the columns, which are expected to further affect the result are preferred. In the research task of this stage, correlation matrices (CM) were used to find columns that only had numeric values.

These correlations indicate the degrees of relationship between the target columns and the potential predictor input columns. The specified correlation value (i.e. ± 0.01) is accepted as the lowest limit in the CMs, so the correlations of the columns with the target columns below this value were ignored for the DM models. During the exploration, the other part of this stage, both numeric and non-numeric columns are shown in histograms to determine visually the reliability or usability of the columns.

In the model development section; the data set was split into training and test data sets. Splitting the data allows the user to create a model and to test this model using data from the same source. Two new training and test tables are created.

Then the main table's rows were allocated to the new training and test tables (i.e. 70 and 30 % of all data set records respectively), by random distribution. SKDS calculates the percentages of the positive and negative values in consultation with the target column (e.g. in Model I: grade point average (GPA) value below 2.0 is negative, above 2.0 positive for each record). A model is then formed using the training data set. Finally, the validity of the model is checked using test data set during the validation process. A lift chart method was chosen for the evaluation of the efficiency of the models. To accomplish all these processes, SKDS benefited from the SQL Server, which is mainly utilized in the data base management activities, and the Analysis Services, which are primarily utilized in modeling and validating.

4.4 Case Study: Modeling Student Performance

This study aims to reveal individual student characteristics that are associated with academic success using a DT classification technique. Each student is categorized as either successful or unsuccessful according to their GPAs. The stages of the study are given below.

4.4.1 Data Description

This study uses the demographic data of students enrolled in the faculty of Economics and Social Sciences of Mugla Sitki Kocman University. This faculty was chosen for this study since the departments are very similar furthermore; it is the oldest in the university and has the most students.

The data used in the discovery process was mostly obtained from when the students registered at the university. The data consisted of; information required by

the state such as city of residence and date of birth, high school information including matriculation certificate, education type and knowledge of foreign language); Turkish university entrance exam score and university placement information, socio-economic status of the student's family and student's academic standing in terms of the semester based GPA scores from their university department.

This study was conducted in the University of Mugla Sitki Kocman, Turkey. In this university, student's academic standing is calculated in the form of the cumulative GPA taking into account all the courses they have taken over the whole degree program. After each course was completed, the student is given a letter grade for which there is a point equivalent (AA = 4.0, BA = 3.5, BB = 3.0, CB = 2.5, CC = 2.0, DC = 1.5, DD = 1.0, FD = 0.5 and FF = 0.0). The GPA is the average of all the grades accumulated over the courses taken within a specific period of education. To be awarded a bachelor degree, a student must obtain at least DD from each course and have a GPA of at least 2.00 to graduate. Honors degrees are given to students with a cumulative GPA from 3.00 to 3.49 and for high honors from 3.50 to 4.00.

4.4.2 Data Preparation

The student data in this study came from many different data files in multiple databases; in the university departments, faculty, central registration system, and archives. Thirteen tables related to the scope of this study were selected from the databases given above. The SQL Server DBMS was implemented at the university registration office. Using SQL queries, the target data set was formed from six separate relational tables, which contained records reflecting academic, demographic, identification, undergraduate and graduate course information. These related tables consisting of a total of 111 columns with 6,470 records were combined in a single view. Afterwards, this target data set is subject to further data cleaning and pre-processing. Unnecessary attributes which are irrelevant for the proposed model are omitted. Thus, handling missing data fields and accounting for known changes is completed, see Fig. 4.4.

The GPAs of the students are generally better indicators because they are reliable and ultimately objective, numeric and accredited to measure academic success in education. In the study, it was found that the greatest correlation coefficient was obtained when comparing the correlations of each column to the other columns of KDD data in the correlation matrices. Thus, the column containing the GPAs was used by the authors as the target column to establish the models. The other columns in the same data set were used as input for the DT models.

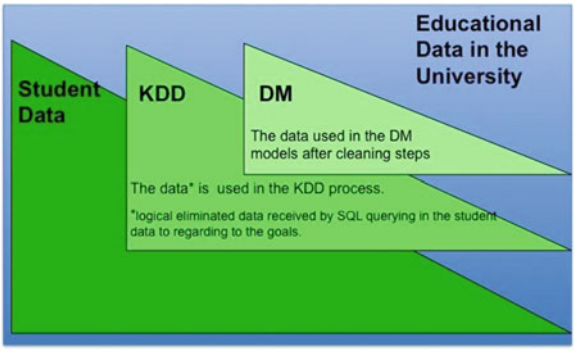

Fig. 4.4 Data selection in the EDM

4.4.3 Analyzer Model

In this study, the classification of the students was undertaken according to their individual success characteristics and DTs were chosen for the model as they assist in producing more understandable results.

Microsoft Decision Trees (MDTs) in the Microsoft SQL Server Analysis Services were applied to create DT models [19]. Originally MDT is a probabilistic classification tree algorithm that is an improvement over the ID3 DT algorithm with some of the add-ons. The basis of the MDT algorithm is introduced in [32].

4.5 Discussion of Results

This section presents the results of the discovery process performed on the student data. Table 4.1 shows the columns and their definitions in the DT models. In the first DT model, the columns affecting the target column, in the order of importance, were YEARECNO and GRANTPTF, which slightly affected the target. The first separations happened in the YEARECNO values: 5, 4, 1 and 2, respectively. This is not a surprising prediction because students have to complete their education over a period of 4–7 years with a requirement of cumulative GPA $\geq$2.00. Thus, the DT models refined as a result of PDCA cycle and YEARECNO were removed from the model to obtain more interesting and hidden results which is a function of DM.

After this operation, the columns that affected the target column are given in the first DT model in Fig. 4.5. Here, the most affective columns used to predict the target column are LANGPREP and REGTYPE. Since LANGPREP is English and REGTYPE is different from the normal type such as transferred from another university, these are seen to be influential on student success in DT. Students transferring from another university must go through adaptation training for a year and must be successful to a certain extent.

Table 4.1 The columns and their definitions for Models 1 and 2

No	Column name	Data included	Data-type	No	Column name	Data included	Data-type
1	KEY	Key column (1, 2, 3, …)	Single	13	LANGPREP	Foreign language to be learned in prep-school	Varchar
2	TARGET_A[a] (target for model I)	If GPA is between 2.0 and 4.0, 1, else 0	Integer	14	PROGTYPE[b]	Department of university	Varchar
3	TARGET_B[b] (target for model II)	If GPA is between 3.0 and 4.0, 1 else 0	Integer	15	HIGHGRAD1[b]	Finishing high school with best average (yes/no)	Integer
4	REQPREP	Request for preparatory school (yes/no)	Boolean	16	SEMRECNO[b]	Number of semesters attended in university	Integer
5	MILITARY	Military service status (completed or not completed)	Boolean	17	SEMCOUNT	Total of semesters to be attended	Integer
6	GENDER	Gender (M/F)	Boolean	18	YEARECNO[a]	The number of years spent at school	Integer
7	GRANTPTF	Grants for tuition fees (receiving/not receiving)	Boolean	19	FMINCOME	Monthly income of family	Single
8	EDUCTYPE	Type of education (day/evening classes)	Boolean	20	CITY2	Region in which the student was born	Varchar
9	DEPTNAME	Department name	Varchar	21	LIVECITY2	Region in which student currently lives	Varchar
10	IDCODE	Department and type of education	Varchar	22	PREFERNO	Order of preference of university attended according to student choice	Integer

(continued)

Table 4.1 (continued)

No	Column name	Data included	Data-type	No	Column name	Data included	Data-type
11	REGTYPE	Type of school registration	Varchar	23	PREFERNO2	Order of preference of university location according to student choice (five categories grouped)	Integer
12	TYPEHIGH	Type of high school	Varchar	24	FMINCOME2[b]	Monthly income of family (four categories grouped)	Integer

[a] only included in the 1st model
[b] only included in the 2nd model

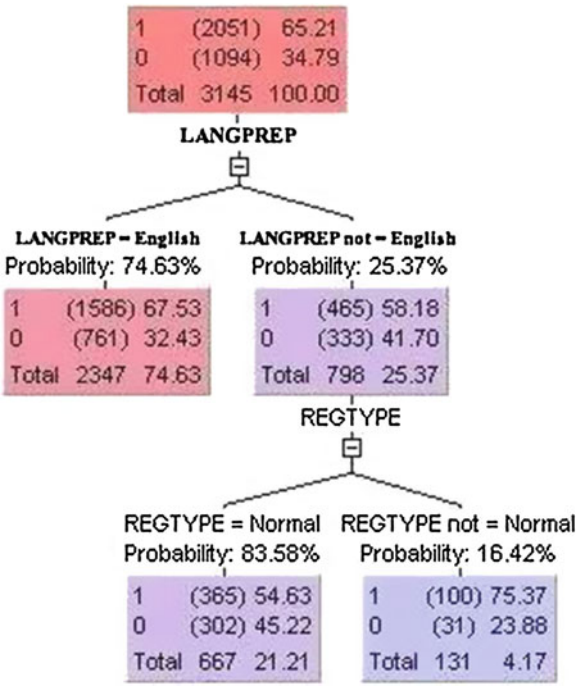

Fig. 4.5 Graphical display of the DTs for Model I

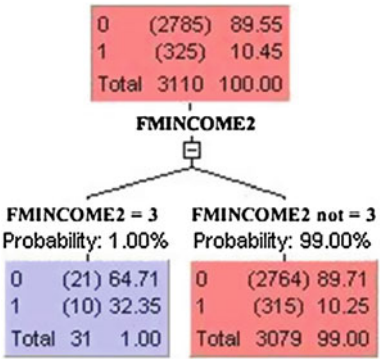

Fig. 4.6 Graphical display of the DTs for Model II

Figure 4.6 shows the resulting DT from Model II. The DT algorithm selected FMINCOME2 as the most important factor that determines the split on the data. The column FMINCOME2 contains categorized data on the monthly incomes of the students' family with the value 3 corresponding to the middle level income (over the lowest rate at which an employer can legally pay an employee; usually

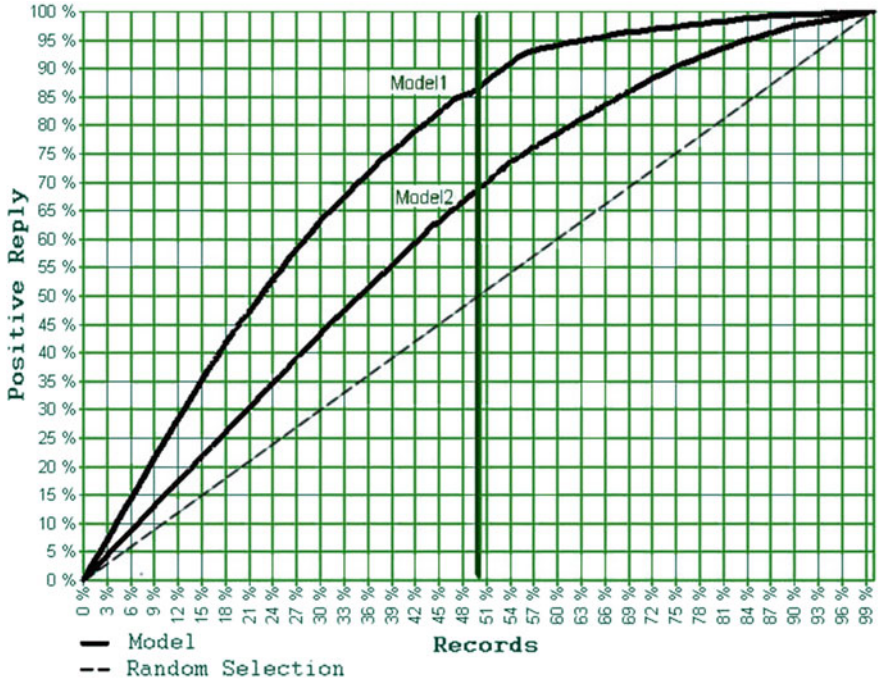

Fig. 4.7 Lift graphics [33] for the models (Model I predicts that the students will have a GPA that is greater or equal to 2.0, whereas Model II predicts that the students will have a GPA that is greater or equal to 3.0)

expressed as pay per month) group. Thus, it is concluded that a high level of success is more correlated with the middle level income.

Lift graphics, which specify the validation of the models are given in Fig. 4.7. When trained models are employed, the percentage of total positive responses in all records was 87 % for Model I and 68 % for Model II. Accordingly, the lift value is 87/50 = 1.74 for Model I and 68/50 = 1.36 for Model II. These results indicate that these models are able to confidently predict the outcome. The reason for the lower prediction potential of Model II is that Model II has less positive values in the target column in comparison with Model I. Thus, a small positive ratio makes it difficult to establish relationships during the training of the model. Model I had a 65 % positive value ratio and for Model II, this ratio was 11 %.

4.6 Conclusions

Besides having a very important role in knowledge production and dissemination, universities offer educational support services for students. Leading universities should discover new ways to base their decision making process in the educational domain on sound business analysis thus providing the best service for their customers; the students. In order to achieve customer satisfaction there needs to be a high student achievement. This can be attained by good guidance and support thus meeting the academic demands of the students during their university education. Analyzing and manipulating existing data with respect to pre-defined goals brings a competitive advantage for higher education institutions in providing high quality, student-specific and student-centered education.

This study aimed to evaluate and develop data-driven approach to the improvement of the performance of university students using a new developed educational technology system, SKDS, using DM methods. With this EDM system, all the tasks involved in the KDP are performed collectively. Our approach has some major advantages for educational institutes. First, the data analysis is realized where it is generated thus, this analysis can be easily repeated on new inserted data. Second, software can use the functions of the SQL server and the Analysis Services, which have essential DM models [34].

In this way, the user can perform tasks without any need for complicated SQL statements. SKDS were designed by the authors considering the needs of the used data and the problem investigated. So, the specifically designed user interface, makes it possible to follow the EDM process and perform the database management activities in an easy and orderly manner. This study may help other researchers working on the integration of specific EDM processes into the DBMS of educational institutions.

In evaluating student performance, the DT classification technique was used since it can produce rules in the tree structure, provide simple and easy-to-understand models and operations that can be carried out even with minimal information preparation. Therefore, DTs can easily be integrated with information

technologies and render high level of automation possible. The classifications attempt to discover which demographic data has the most impact on student GPA. In the current study two classification models were obtained limited to determining the profiles of students whose GPAs ranged from 2.0 to 4.0 and the second group with GPAs of 3.0–4.0.

In the first model, the types of registration to the university and in the second model, the monthly income of the family were found to be the greatest factors affecting the target. In checking the performance of the models, lift graphics were used. According to the lift graphics, values 1.74 for Model I and 1.36 for Model II were found which shows that the models have a prediction capability.

Missing data in some of the columns in the dataset had a direct impact on the success of the system. Therefore, more accurate predictions about student success can be made when the amount of data and the number of variables is increased. DTs handle non-numeric data very well. This ability to accept categorical data minimizes the amount of data transformations. Even though DTs are good at classifying data, they alone may not be sufficient for discovery.

In terms of future work, since SKDS can connect to the SQL Server and the Analysis Services, which include more DM algorithms, an extension of the current study based on different analysis parameters could show different perspectives of a student's performance and progress through their university career.

References

1. Abdous, M., He, W., Yen, C.J.: Using data mining for predicting relationships between online question theme and final grade. J. Educ. Technol. Soc. **15**(3), 77–88 (2012)
2. Campagni R., Merlini D., Sprugnoli R.: Analyzing paths in a student database. In: Yacef, K., Zaïane, O., Hershkovitz, A., Yudelson, M., Stamper, J. (eds.) 5th International Conference on Educational Data Mining, pp. 208–209. International Educational Data Mining Society, Chania (2012)
3. Oyelade, O.J., Oladipupo, O.O., Obagbuwa, I.C.: Application of k-means clustering algorithm for prediction of students' academic performance. Int. J. Comput. Sci. Inf. Secur. **7**(1), 292–295 (2010)
4. Scheuer, O., McLaren, B.M.: Educational DM. In: Seel, N.M. (ed.) Encyclopedia of the Sciences of Learning. Springer, New York (2011)
5. Bidgoli B.M., Kashy D.A., Kortemeyer G., Punch W.F.: Predicting student performance: an application of DM methods with an educational web-based system. In: 33rd ASEE/IEEE Frontiers in Education Conference, pp. 13–18. IEEE, Boulder (2003)
6. Romero, C., Ventura, S.: Educational data mining: a review of the state of the art. IEEE Trans. Syst. Man Cybern. Part C **40**(6), 601–618 (2010)
7. Delavari, N., Amnuaisuk, S.P., Beikzadeh, M.R.: DM application in higher learning institutions. Inform. Educ. **7**(1), 31–54 (2008)
8. Vialardi, C., Chue, J., Peche, J.P., Alvarado, G., Vinatea, B., Estrella, J., Ortigosa, A.: A data mining approach to guide students through the enrollment process based on academic performance. User Model. User-Adap. Inter. **21**(1–2), 217–248 (2011)
9. Kumar, N.V.A., Uma, G.V.: Improving academic performance of students by applying data mining technique. Eur. J. Sci. Res. **34**(4), 526–534 (2009)

10. IBM Case Study, Hamilton County Department of Education: Improving student performance and school effectiveness with predictive analytics. http.//www.ibm.com/analytics/us/en/case-studies
11. IBM Case Study, Seton Hall University: Social media marketing analytics helps engage incoming prospects and increase enrollment yield http://www.ibm.com/analytics/us/en/case-studies
12. Cios, K.J., Swiniarski, R.W., Pedrycz, W., Kurgan, L.A.: Data Mining: A Knowledge Discovery Approach, pp. 9–24. Springer, New York (2007)
13. Meints, M., Möller, J.: Privacy preserving data mining: a process centric view from a European perspective. Report of the project FIDIS (Future of Identity in the Information Society), http://www.fidis.net/fileadmin/journal/issues/1-2007/Privacy_Preserving_Data_Mining.pdf
14. Jalili, M., Rezaie, K.: Quality principles deployment to achieve strategic results. Int. J. Bus. Excellence **3**(2), 226–259 (2010)
15. Maimon, O., Rokach, L.: Introduction to knowledge discovery and data mining. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 1–15. Springer, New York (2010)
16. Mićić, Ž., Mićić, M., Blagojević, M.: ICT Innovations at the platform of standardisation for knowledge quality in PDCA. Comput. Stand. Interfaces **36**(1), 231–243 (2013)
17. Guruler, H., Istanbullu, A., Karahasan, M.: A new student performance analyzing system using knowledge discovery in higher educational databases. Comput. Educ. **55**(1), 247–254 (2010)
18. Guruler, H., Karahasan, M., Istanbullu, A.: Determining profile of university students: a case study on Mugla University databases. Mugla Univ. J. Soc. Sci. **18**, 27–37 (2007)
19. Larson, B., English, D., Purington, P.: Delivering Business Intelligence with Microsoft SQL Server 2012. McGraw-Hill, New York (2012)
20. The Cyber Security and Information Systems Information Analysis Center (CSIAC): A Comparison of Leading DM Tools. https://sw.thecsiac.com/databases/url/key/222/225
21. Chikalov, I., Lozin, V., Lozina, I., Moshkov, M., Nguyen, H.S., Skowron, A., Zielosko, B.: Logical analysis of data: theory, methodology and applications. In: Chikalov, I., Lozin, V., Lozina, I., Moshkov, M., Nguyen, H.S., Skowron, A., Zielosko, B. (eds.) Three Approaches to Data Analysis, pp. 147–192. Springer, Heidelberg (2013)
22. Khan, D.M., Mohamudally, N., Babajee, D.K.R.: A unified theoretical framework for data mining. Procedia Comput. Sci. **17**, 104–113 (2013)
23. Özekes, S.: Classification and prediction in data mining with neural networks. Istanbul Univ. J. Electr. Electron. Eng. **3**(1), 707–712 (2012)
24. Cano, A., Zafra, A., Ventura, S.: An interpretable classification rule mining algorithm. Inf. Sci. **240**, 1–20 (2013)
25. Guan, H.: A new data mining approach combining with extension transformation of extenics. In: Deng, W. (ed.) Future Control and Automation, vol. 173, pp. 199–205. LNEESpringer, Heidelberg (2012)
26. Lakshmi, T.M., Martin, A., Begum, R.M., Venkatesan, V.P.: An analysis on performance of decision tree algorithms using student's qualitative data. Int. J. Mod. Educ. Comput. Sci. **5**(5), 18–27 (2013)
27. Lin, C.F., Yeh, Y.C., Hung, Y.H., Chang, R.I.: Data mining for providing a personalized learning path in creativity: an application of decision trees. Comput. Educ. **68**, 199–210 (2013)
28. James, G., Witten, D., Hastie, T., Tibshirani, R.: Tree-based methods. In: Casella, G., Fienberg, S., Olkin, I. (eds.) An Introduction to Statistical Learning, vol. 41, pp. 303–335. Springer, New York (2013)
29. Yang, H., Fong, S.: Optimized very fast decision tree with balanced classification accuracy and compact tree size. In: 3rd International Conference on Data Mining and Intelligent Information Technology Applications, pp. 57–64. IEEE, Coloane (2011)
30. López-Chau, A., Cervantes, J., López-García, L., García Lamont, F.: Fisher's Decision Tree. Expert Systems with Applications **40**(16), 6283–6291 (2013)

31. Aggarwal, C.C., Zhai, C.: A survey of text classification algorithms. In: Aggarwal, C.C., Zhai, C.X. (eds.) Mining Text Data, pp. 163–222. Springer, New York (2012)
32. Microsoft Decision Trees Algorithm Technical Reference http://msdn.microsoft.com/en-us/library/cc645868.aspx
33. Prati, R.C., Batista, G.E.A.P.A., Monard, M.C.: A survey on graphical methods for classification predictive performance evaluation. IEEE Trans. Knowl. Data Eng. **23**(11), 1601–1618 (2011)
34. Data Mining Algorithms (Analysis Services - Data Mining) http://msdn.microsoft.com/en-us/library/ms175595.aspx