CHAPTER 1

EDUCATIONAL PROCESS MINING: A TUTORIAL AND CASE STUDY USING MOODLE DATA SETS

Cristóbal Romero[1], Rebeca Cerezo[2], Alejandro Bogarín[1], and Miguel Sánchez-Santillán[2]

[1] Department of Computer Science, University of Córdoba, Córdoba, Spain

[2] Department of Psychology, University of Oviedo, Oviedo, Spain

The use of learning management systems (LMSs) has grown exponentially in recent years, which has had a strong effect on educational research. An LMS stores all students' activities and interactions in files and databases at a very low level of granularity (Romero, Ventura, & García, 2008). All this information can be analyzed in order to provide relevant knowledge for all stakeholders involved in the teaching–learning process (students, teachers, institutions, researchers, etc.). To do this, data mining (DM) can be used to extract information from a data set and transform it into an understandable structure for further use. In fact, one of the challenges that the DM research community faces is determining how to allow professionals, apart from computer scientists, to take advantage of this methodology. Nowadays, DM techniques are applied successfully in many areas, such as business marketing, bioinformatics, and education. In particular, the area that applies DM techniques in educational settings is called educational data mining (EDM). EDM deals with unintelligible, raw educational data, but one of the core goals of this discipline—and the present chapter—is to make this valuable data legible and usable to students as feedback, to professors as assessment, or to universities for strategy. EDM is broadly studied, and a reference tutorial was developed by Romero et al. (2008). In this tutorial, the authors show the step-by-step process for doing DM with Moodle data. They describe how to apply preprocessing and traditional DM techniques

Data Mining and Learning Analytics: Applications in Educational Research, First Edition.
Edited by Samira ElAtia, Donald Ipperciel, and Osmar R. Zaïane.

(such as statistics, visualization, classification, clustering, and association rule mining) to LMS data.

One of the techniques used in EDM is process mining (PM). PM starts from data but is process centric; it assumes a different type of data: events. PM is able to extract knowledge of the event log that is commonly available in current information systems. This technique provides new means to discover, monitor, and improve processes in a variety of application domains. The implementation of PM activities results in models of business processes and historical information (more frequent paths, activities less frequently performed, etc.). Educational process mining (EPM) involves the analysis and discovery of processes and flows in the event logs generated by educational environments. EPM aims to build complete and compact educational process models that are able to reproduce all the observed behaviors, check to see if the modeling behavior matches the behavior observed, and project extracted information from the registrations in the pattern to make the tacit knowledge explicit and to facilitate a better understanding of the process (Trcka & Pechenizkiy, 2009).

EPM has been previously applied successfully to the educational field; one of the most promising applications is used to study the difficulties that students of different ages show when learning in highly cognitively and metacognitively demanding learning environments, such as a hypermedia learning environment (Azevedo et al., 2012). These studies describe suppositions and commonalities across several of the foremost EPM models for self-regulated learning (SRL) with student-centered learning environments (SCLEs). It supplies examples and definitions of the key metacognitive monitoring processes and the regulatory skills used when learning with SCLEs. It also explains the assumptions and components of a leading information processing model of SRL and provides specific examples of how EPM models of metacognition and SRL are embodied in four current SCLEs.

However, several problems have been previously found when using EPM (Bogarín et al., 2014). For instance, the model obtained is not well adjusted to the general behavior of students, and the resulting model may be too large and complex for a teacher or student to analyze. In order to solve these problems, we propose the use of clustering for preprocessing the data before applying EPM to improve understanding of the obtained models. Clustering techniques divide complex phenomena—described by sets of objects or by highly dimensional data—into small, comprehensible groups that allow better control and understanding of information. In this work, we apply clustering as a preprocessing task for grouping users based on their type of course interactions. Thus, we expect to discover the most specific browsing behaviors when using only the clustered data rather than the full data set. This chapter describes, in a practical tutorial, how to apply clustering and EPM to Moodle data using two well-known open-source tools: Weka (Witten, Frank, & Hall, 2011) and ProM (Van der Aalst, 2011a).

The chapter is organized as follows: Section 1.1 describes the most relevant works related to the chapter, Section 1.2 describes the data preparation and clustering, Section 1.3 describes the application of PM, and Section 1.4 outlines some conclusions and suggestions for further research.

1.1 BACKGROUND

Process mining (PM) is a data mining (DM) technique that uses event logs recorded by systems in order to discover, monitor, and improve processes in different domains. PM is focused on processes, but it also uses the real data (Van der Aalst, 2011a). It is the missing link between the classical process model of analysis and data-oriented analysis like DM and machine learning. We can think of PM as a bridge between processes and data, between business process management and business intelligence, and between compliance and performance. PM connects many different ideas, and that makes it extremely valuable (Van der Aalst, 2011b).

The starting point for PM is event data. We assume that there is an event log in which each event refers to a case, an activity, and a point in time or time stamp. An event log can be seen as a collection of cases (which we sometimes also refer to as traces); each case corresponds to a sequence of events. Event data comes from a large variety of sources. PM consists of different types of mining (Van der Aalst et al., 2012):

- **Process discovery** conforms to a model.
- **Conformance checking** is a form of replay aimed at finding deviations.
- **Enhancement** is also a form of replay with the goal of finding problems (such as bottlenecks) or ideas for improvement.

The potential and challenges of PM have been previously investigated in the field of professional training (Cairns et al., 2014). For instance, this field has focused on the mining and analysis of social networks involving course units or training providers; it has also proposed a two-step clustering approach for partitioning educational processes following key performance indicators. Sedrakyan, Snoeck, and De Weerdt (2014) attempted to obtain empirically validated results for conceptual modeling of observations of activities in an educational context. They tried to observe the characteristics of the modeling process itself, which can be associated with better/worse learning outcomes. In addition, the study provided the first insights for learning analytics research in the domain of conceptual modeling.

The purpose of another interesting study, which was conducted by Schoor and Bannert (2012), was to explore sequences of social regulatory processes during a computer-supported collaborative learning task and to determine these processes' relationship to group performance. Using an analogy to self-regulation during individual learning, the study conceptualized social regulation as both individual and collaborative activities: analyzing, planning, monitoring, and evaluating cognitive and motivational aspects during collaborative learning. In an exploratory way, the study used PM to identify process patterns for high and low group performance dyads.

Referring to the research on self-regulated learning (SRL), the recent work of Bannert, Reimann, and Sonnenberg (2014) analyzed individual regulation in terms of a set of specific sequences of regulatory activities. Thus, the aim of the study's approach was to analyze the temporal order of spontaneous individual regulation activities. This research demonstrates how various methods developed in the PM

research can be applied to identify process patterns in SRL events, as captured in verbal protocols. It also shows how theoretical SRL process models can be tested with PM methods.

Another related work observed how linking labels in event logs to their underlying semantics can bring educational process discovery to the conceptual level (Cairns et al., 2004). In this way, more accurate and compact educational processes can be mined and analyzed at different levels of abstraction. It is important to say that this approach was done using the ProM framework (Van der Aalst, 2011a).

ProM contains the Heuristics Miner plug-in, which has been used to analyze a student's written activities and thus to improve the student's writing skills. Southavilay, Yacef, and Calvo (2010) presented a job that enables the development of a basic heuristic to extract the semantic meaning of text changes and determine writing activities. Heuristics have been able to analyze the activities of student writing using PM and have found patterns in these activities. The discovered patterns, the snapshot of processes provided by the sequence of action, and the dotted chart analysis can be used to provide feedback to students so that they are aware of their writing activities. One way to improve understanding of how writing processes lead to a better outcome is to improve heuristics (Boiarsky, 1984). In this work, only changes in spelling, numbers, ratings, and formats are considered. No grammatical corrections are included. In addition, one of the proposed changes is vocabulary improvement. Another concept that is not taken into account in this work is the repetition of words; good writers often avoid the annoying repetition of words and instead use synonyms. Finally, the Heuristic Miner was previously used to investigate the processes recorded by students at the University of Thailand to minimize the educational adaptation process (Ayutaya, Palungsuntikul, & Premchaiswadi, 2012). The referenced work demonstrated the behavior of Heuristic Miner in the extraction of a slightly structured process. The properties of the Heuristics Miner plug-in were shown using an event log from the University of Thailand. In addition, the Heuristics Miner was also used to analyze learning management system (LMS) learning routes and to track the behavior learned in relation to the respective learning styles, which must be identified in advance.

On the other hand, the process of grouping students is also very relevant for educational data mining (EDM). This naturally refers to an area of data analysis, namely, data clustering, which aims to discover the natural grouping structure of a data set. A pair of good reviews was conducted about the application of clustering techniques for improving e-learning environments. The first review, by Vellido, Castro, and Nebot (2011), was devoted to clustering educational data and its corresponding analytical methods; these methods hold the promise of providing useful knowledge to the community of e-learning practitioners. The authors of this review described clustering and visualization methods that enhance the e-learning experience due to the capacity of the former to group similar actors based on similarities and the ability of the latter to describe and explore these groups intuitively.

The second review, by Dutt et al. (2015), aimed to consolidate the different types of clustering algorithms applied in the EDM context and to answer the question of how a higher educational institution can harness the power of didactic data for strategic use. Building an information system that can learn from data is a difficult

task, but it has been achieved using various data mining approaches, such as clustering, classification, and prediction algorithms.

Finally, we want to note that we found no works that use clustering techniques together with educational process mining (EPM). Thus, with the exception of our previous works (Bogarín et al., 2014), to our knowledge, there have been no published works about this topic. In fact, this chapter is an in-depth extension of our previous short work (Bogarín et al., 2014); we have reoriented this chapter to be a practical guide that can also be used by a nonexpert, such as an instructor.

1.2 DATA DESCRIPTION AND PREPARATION

The data sets used in this work were gathered from a Moodle 2.0 course used by 84 undergraduate students from the psychology degree program at a university in northern Spain. The experiment was implemented during two semesters as an assignment for a third-year compulsory subject. Students were asked to participate in an e-learning/training program about "learning to learn" and a SRL that was to be completed entirely outside of teaching hours. The program was made up of 11 different units that were sent to the students on a weekly basis, and each student was able to work on each unit for a 15-day period. Students got an extra point on their final subject grade if they completed at least 80% of the assignments.

1.2.1 Preprocessing Log Data

Moodle logs every click that students make for navigational purposes (Van der Aalst et al., 2012). Moodle has a modest built-in log-viewing system (see Fig. 1.1). Log files can be filtered by course, participant, day, and activity, and they can be shown

Course	IP Address	Date	Full name	Action	Information
Trastornos del Apre	156.35.71.136	2-10-2012-12:35	FERNANDEZ MARTINEZ Carla	course view	Trastornos del Aprendizaje (Grado en Psicología)
Trastornos del Apre	81.9.215.5	2-10-2012-12:58	ALVAREZ SAN MILLAN Andrea	course view	Trastornos del Aprendizaje (Grado en Psicología)
Trastornos del Apre	81.9.215.5	2-10-2012-12:58	ALVAREZ SAN MILLAN Andrea	questionnaire view	PROYECTO E-TRAL
Trastornos del Apre	156.35.221.243	2-10-2012-13:30	HOMBACH VIOLA MARIANNA	course view	Trastornos del Aprendizaje (Grado en Psicología)
Trastornos del Apre	156.35.221.243	2-10-2012-13:32	HOMBACH VIOLA MARIANNA	page view	Hoja de Ruta
Trastornos del Apre	156.35.221.243	2-10-2012-13:32	HOMBACH VIOLA MARIANNA	folder view all	
Trastornos del Apre	156.35.221.243	2-10-2012-13:32	HOMBACH VIOLA MARIANNA	resource view all	
Trastornos del Apre	156.35.221.243	2-10-2012-13:32	HOMBACH VIOLA MARIANNA	label view all	
Trastornos del Apre	156.35.221.243	2-10-2012-13:32	HOMBACH VIOLA MARIANNA	page view all	
Trastornos del Apre	156.35.221.243	2-10-2012-13:32	HOMBACH VIOLA MARIANNA	imscp view all	
Trastornos del Apre	156.35.221.243	2-10-2012-13:32	HOMBACH VIOLA MARIANNA	url view all	
Trastornos del Apre	83.97.248.62	2-10-2012-13:51	CARRIO CARRO Luis	course view	Trastornos del Aprendizaje (Grado en Psicología)
Trastornos del Apre	93.156.24.124	2-10-2012-14:16	Rodríguez Carballo Andrea	course view	Trastornos del Aprendizaje (Grado en Psicología)
Trastornos del Apre	156.35.221.243	2-10-2012-14:23	HOMBACH VIOLA MARIANNA	page view	Carta cero
Trastornos del Apre	156.35.221.243	2-10-2012-15:05	HOMBACH VIOLA MARIANNA	label view all	
Trastornos del Apre	156.35.221.243	2-10-2012-15:05	HOMBACH VIOLA MARIANNA	imscp view all	
Trastornos del Apre	156.35.221.243	2-10-2012-15:05	HOMBACH VIOLA MARIANNA	resource view all	
Trastornos del Apre	156.35.221.243	2-10-2012-15:05	HOMBACH VIOLA MARIANNA	url view all	
Trastornos del Apre	156.35.221.243	2-10-2012-15:05	HOMBACH VIOLA MARIANNA	page view all	
Trastornos del Apre	156.35.221.243	2-10-2012-15:05	HOMBACH VIOLA MARIANNA	folder view all	
Trastornos del Apre	80.39.86.208	2-10-2012-15:16	Sánchez Sánchez María	course view	Trastornos del Aprendizaje (Grado en Psicología)
Trastornos del Apre	88.29.14.156	2-10-2012-15:23	García Pérez LAURA	course view	Trastornos del Aprendizaje (Grado en Psicología)
Trastornos del Apre	88.29.14.156	2-10-2012-15:24	García Pérez LAURA	forum view forum	Novedades
Trastornos del Apre	88.29.14.156	2-10-2012-15:24	García Pérez LAURA	forum view forum	Tablón de Anuncios
Trastornos del Apre	88.29.14.156	2-10-2012-15:24	García Pérez LAURA	page view	Hoja de Ruta
Trastornos del Apre	88.29.14.156	2-10-2012-15:25	García Pérez LAURA	page view	Carta cero
Trastornos del Apre	156.35.71.136	2-10-2012-15:34	Alonso Vega Jesus	course view	Trastornos del Aprendizaje (Grado en Psicología)

Figure 1.1 Moodle event log.

or saved in files with the formats: text format (TXT), open document format for office applications (ODS), or Microsoft excel file format (XLS).

We did not use all the information included in the Moodle log file provided (see Table 1.1). In particular, we did not use the name of the course (because it is the same for all records) or the internet protocol (IP) address (because it is irrelevant for our purposes).

Additionally, we have also filtered the log file to eliminate those records that contain an action that could be considered irrelevant to the students' performance. Thus, from all the actions that Moodle stored in our log file (39 in total), we only used the 20 actions that were related to the students' activities in the course (see Table 1.2). This filter lets us reduce the log file from 41,532 to 40,466 records.

Then, we created a new attribute by joining the action and information attributes. We implemented this transformation because it provides additional valuable

TABLE 1.1 Variables of the Moodle log file

Attribute	Description
Course	The name of the course
IP address	The IP of the device used to access
Time	The date they accessed it
Full name	The name of the student
Action	The action that the student has done
Information	More information about the action

TABLE 1.2 Actions considered relevant to the students' performance

assignment upload
assignment view
course view
folder view
forum add discussion
forum add post
forum update post
forum view discussion
forum view forum
page view
questionnaire submit
questionnaire view
quiz attempt
quiz close attempt
quiz continue attempt
quiz review
quiz view
quiz view summary
resource view
url view

TABLE 1.3 List of events in the quiz view after joining action and information

quiz view: Actividad 11
quiz view: Actividad 4
quiz view: Actividad 6
quiz view: Actividad 7
quiz view: Actividad 9
quiz view: Actividad Mapa Conceptual
quiz view: Actividad Tema 2
quiz view: Actividad Tema 3 El Código Secreto
quiz view: Actividad Tema 3 Toma de apuntes
quiz view: Carta 1
quiz view: Carta 10
quiz view: Carta 11
quiz view: Carta 2
quiz view: Carta 3
quiz view: Carta 4
quiz view: Carta 5
quiz view: Carta 6
quiz view: Carta 7
quiz view: Carta 8
quiz view: Carta 9
quiz view: Neutra II
quiz view: Neutra III
quiz view: Subrayado y resumen
quiz view: Tarea neutra enfermedad
quiz view: Tarea: Aprende a Relajarte

information related to the action. For example, a particular action in the quiz view was associated with 25 different information fields, as shown in Table 1.3 (action: information). After completing this transformation, we obtained a total of 332 events (actions plus the information field) that students executed when browsing the course.

Finally, it was necessary to transform the files into the appropriate format for use by the ProM (Van der Aalst, 2011a) tool. To do this, the Moodle log file was firstly saved in the comma-separated values (CSV) format, as shown in Figure 1.2.

Then, the CSV file was converted to mining extensible markup language (MXML), which is the format interpreted by ProM. We used the ProM Import Framework to do this conversion. We selected the option "General CSV File" from the "Filter" properties tab (see Fig. 1.3), and we linked the names of the head of this CSV file with corresponding labels in the properties panel:

- The "Case ID" property was linked with the "Action" value.
- The "Task ID" property was linked with the "Information" value.
- The "Start Time" property was linked with the "Time" value.
- The "Originator" property was linked with the "Full Name" value.

It is also important to set the "Date Format" field correctly; in this case, the format is "D-M-Y-H: M."

total.csv

```
1  Time;FullName;Action;Information
2  10-10-2012-19:28;Pisatti Combina Santiago Matias;course view;course view
3  10-10-2012-19:28;Pisatti Combina Santiago Matias;forum view discussion;forum view discussion
4  10-10-2012-19:28;Pisatti Combina Santiago Matias;forum view forum;forum view forum
5  10-10-2012-19:40;Pisatti Combina Santiago Matias;forum view discussion;forum view discussion
6  10-11-2012-18:26;Pisatti Combina Santiago Matias;course view;course view
7  10-11-2012-18:26;Pisatti Combina Santiago Matias;quiz attempt;quiz attempt
8  10-11-2012-18:26;Pisatti Combina Santiago Matias;quiz continue attempt;quiz continue attempt
9  10-11-2012-18:26;Pisatti Combina Santiago Matias;quiz view;quiz view
10 10-11-2012-18:32;Pisatti Combina Santiago Matias;course view;course view
11 10-11-2012-18:32;Pisatti Combina Santiago Matias;quiz close attempt;quiz close attempt
12 10-11-2012-18:32;Pisatti Combina Santiago Matias;quiz review;quiz review
13 10-11-2012-18:32;Pisatti Combina Santiago Matias;quiz view summary;quiz view summary
14 10-11-2012-18:32;Pisatti Combina Santiago Matias;resource view;resource view
15 10-1-2013-16:26;Pisatti Combina Santiago Matias;course view;course view
16 10-1-2013-16:27;Pisatti Combina Santiago Matias;folder view;folder view
17 11-10-2012-19:55;Pisatti Combina Santiago Matias;course view;course view
18 11-10-2012-19:56;Pisatti Combina Santiago Matias;forum view discussion;forum view discussion
19 11-10-2012-19:56;Pisatti Combina Santiago Matias;forum view forum;forum view forum
20 11-10-2012-19:58;Pisatti Combina Santiago Matias;forum view discussion;forum view discussion
21 1-11-2012-20:34;Pisatti Combina Santiago Matias;course view;course view
22 1-11-2012-20:34;Pisatti Combina Santiago Matias;forum view discussion;forum view discussion
23 1-11-2012-20:34;Pisatti Combina Santiago Matias;forum view forum;forum view forum
24 11-1-2013-19:58;Pisatti Combina Santiago Matias;course view;course view
25 11-1-2013-19:58;Pisatti Combina Santiago Matias;questionnaire view;questionnaire view
26 1-1-2013-18:31;Pisatti Combina Santiago Matias;course view;course view
27 12-11-2012-16:04;Pisatti Combina Santiago Matias;course view;course view
28 12-11-2012-16:04;Pisatti Combina Santiago Matias;quiz view;quiz view
29 12-11-2012-20:41;Pisatti Combina Santiago Matias;course view;course view
```

Figure 1.2 Moodle event log in CSV format.

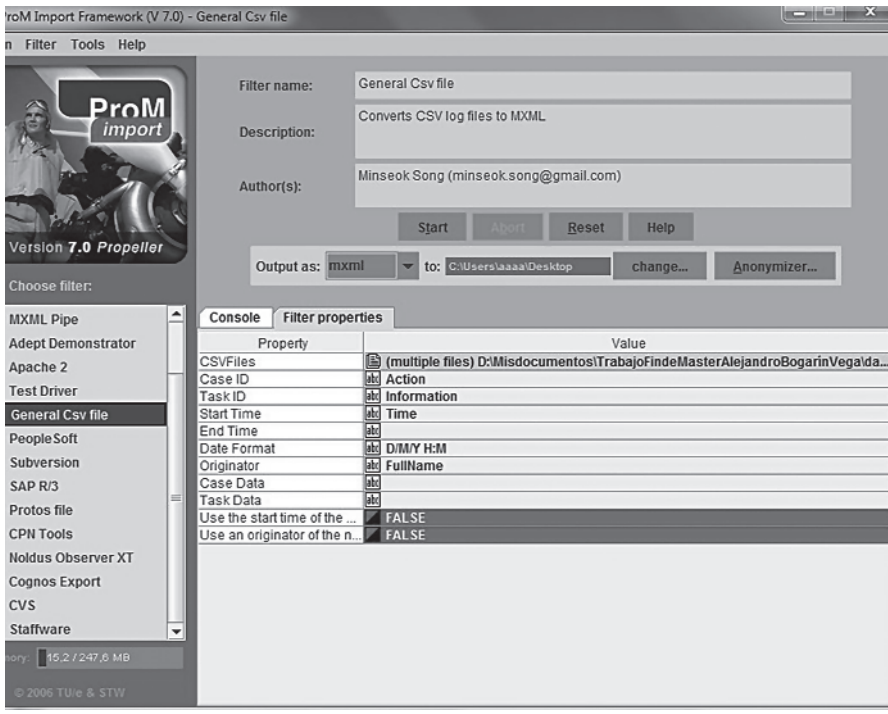

Figure 1.3 Interface for the ProM import tool.

total.mxml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- MXML version 1.1 -->
<!-- Created by ProM Import Framework, Version 7.0 (Propeller) -->
<!-- via MXMLib Version 1.9 (http://promimport.sf.net/) -->
<!-- (c) 2004-2007 C.W. Guenther (christian@deckfour.org); Eindhoven Technical University -->
<!-- This event log is formatted in MXML, for use by BPI and Process Mining Tools. -->
<!-- You can load this file e.g. in the ProM Framework for Process Mining. -->
<!-- More information about MXML, Process Mining, and ProM: http://www.processmining.org/. -->
<WorkflowLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://is.tm.tue.nl/research/processmining/WorkflowLog.xsd"
description="Unified single process">
    <Data>
        <Attribute name="app.name">ProM Import Framework</Attribute>
        <Attribute name="app.version">7.0 (Propeller)</Attribute>
        <Attribute name="java.vendor">Oracle Corporation</Attribute>
        <Attribute name="java.version">1.7.0_21</Attribute>
        <Attribute name="mxml.creator">MXMLib (http://promimport.sf.net/)</Attribute>
        <Attribute name="mxml.version">1.1</Attribute>
        <Attribute name="os.arch">x86</Attribute>
        <Attribute name="os.name">Windows XP</Attribute>
        <Attribute name="os.version">5.1</Attribute>
        <Attribute name="user.name">alex</Attribute>
    </Data>
    <Source program="CSV files"/>
    <Process id="UNIFIED" description="Unified single process">
        <ProcessInstance id="assignment view">
            <AuditTrailEntry>
                <WorkflowModelElement>assignment view</WorkflowModelElement>
                <EventType>start</EventType>
                <Timestamp>2012-10-15T09:00:00.000+02:00</Timestamp>
                <Originator>HOMBACH VIOLA MARIANNA</Originator>
            </AuditTrailEntry>
            <AuditTrailEntry>
                <WorkflowModelElement>assignment view</WorkflowModelElement>
                <EventType>complete</EventType>
                <Timestamp>2012-10-15T09:00:00.000+02:00</Timestamp>
                <Originator>HOMBACH VIOLA MARIANNA</Originator>
            </AuditTrailEntry>
```

Figure 1.4 MXML file for use with ProM.

The file resulting from the filter is shown in Figure 1.4. This file is then used with ProM in order to do EPM.

1.2.2 Clustering Approach for Grouping Log Data

We also propose an approach for using clustering as a preprocessing task for improving EPM. The traditional approach uses all event log data to disclose a process model of a student's behavior. However, this approach applies clustering first in order to group students with similar marks or characteristics; then, it implements PM to discover more specific models of the student's behavior (see Fig. 1.5).

The proposed approach used two clustering/grouping methods:

1. *Manual clustering*: grouping students directly using only the students' marks on the course's final exam.
2. *Automatic clustering*: grouping students using a clustering algorithm based on their interactions with the Moodle course.

Manual clustering uses the student's final mark, which is a numeric value on a 10-point scale provided by the instructor. We turned this continuous value into a

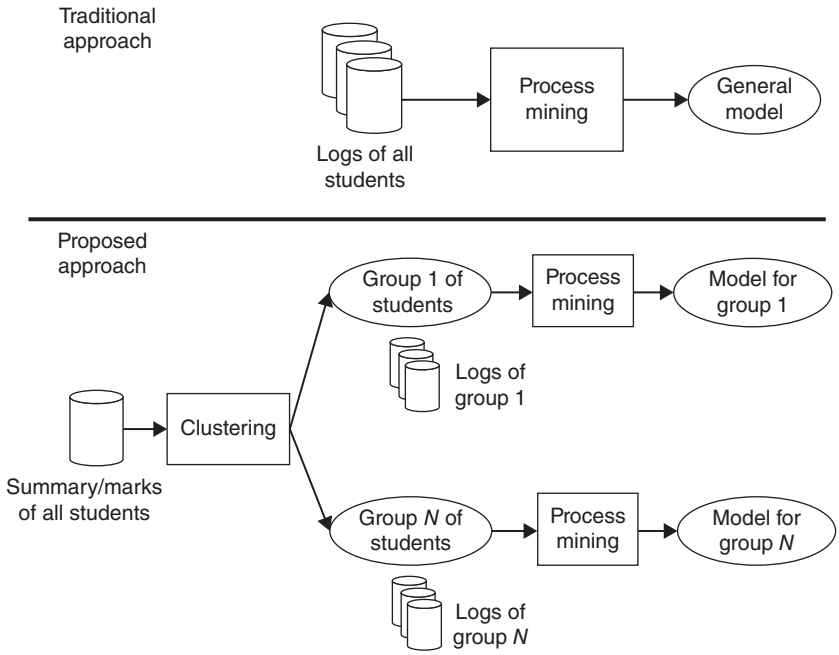

Figure 1.5 Representation of the proposed approach versus the traditional approach.

categorical value using Spain's traditional academic grading system: *fail* (from 0 to 4.9) and *pass* (from 5 to 10). By applying this manual clustering approach, two groups are easily detected from the 84 students:

- 16 students whose final marks were less than 5 (*fail*)
- 68 students whose final marks were greater than or equal to 5 (*pass*)

Automatic clustering uses the Moodle usage data, which were obtained after students worked on the course. We mainly used the reports or summaries of each student's interactions in Moodle. It is important to note that we have only used selected variables and that we have filtered the actions in our log file. The variables selected can be grouped into four different types (see Tables 1.4, 1.5, 1.6, and 1.7) by taking into account what they represent on a higher granularity level: variables related to time spent working, variables related to procrastination, variables related to participation in forums, and other variables.

This LMS version stores a total of 76 variables, but looking at previous works in the same line, only 12 actions make sense when representing the students' performance in the Moodle course used for the experiment. Some of these actions are extracted directly from Moodle records; however, to make sense of the data, it is sometimes advisable to formulate queries to obtain aggregated results. Therefore, other variables are calculated based on those records using a simple operation; for example, as seen in Table 1.5, the variable *days task* is calculated by subtracting the date on which the student uploaded the task (Moodle records this automatically from

TABLE 1.4 Variables related to the time spent working

Name	Description	Extraction Method and Moodle Nomenclature	Additional Information
Time theory	Total time spent on theoretical components of the content	Sum of the periods between resource view and the next different action	Students have a period of 15 days to learn from theory. The number of units is designed to be implemented during one semester on a weekly basis. Students have available one different unit every Monday. The theoretical contents remain available gradually till the end of the program
Time task	Total time spent in instructional tasks	Sum of the periods between *quiz view*/*quiz attempt*/*quiz continue attempt*/*quiz close attempt* and the next different action	Students have a period of 15 days to complete the tasks. The number of units is designed to be implemented during one semester on a weekly basis. A period of 15 days is imposed to do the task in order to coregulate them and avoiding procrastination
Time forums	Total time spent reviewing forums	Sum of the periods between *forum view* and the next different action	Students have a period of 15 days to go through their peers' comments and post at the forums

TABLE 1.5 Variables related to procrastination

Name	Description	Extraction Method and Moodle Nomenclature	Additional Information
Days theory	How many days in a 15-day period the students wait to check the content at least once (in days)	Date of *resource view* since the content is available	Students have a period of 15 days to learn from texts
Days tasks	How many days in a 15-day period the students wait to check the task at least once (in days)	Date of *task view* since the task is available	Students have a period of 15 days to complete the tasks
Days "hand in"	How many days in a 15-day period take the students to complete the task (in days)	Date of *quiz close attempt* since the task is available	Students have a period of 15 days to complete the tasks
Days forum	How many days in a 15-day period the students wait to check the forum (in days)	Date of *forum view forum* or *forum view discussion* since the forum is available	Students have a period of 15 days to go through their peers' comments
Days posting	How many days in a 15-day period take the students to post in the forum (in days)	Date of *forum add discussion* or *forum add replay* since the forum is available	Students have a period of 15 days to post at the forums

TABLE 1.6 Variables related to participation in forums

Name	Description	Extraction Method and Moodle Nomenclature	Additional Information
Words fórums	Number of words in forum posts	Extracting the number of *forum add discussion* and *forum add replay* words	There is no restriction in the number of words that students can use in the post
Sentences fórums	Number of sentences in forum posts	Extracting the number of *forum add discussion* and *forum add replay* sentences	There is no restriction in the number of sentences that students can use in the post

TABLE 1.7 Other variables

Name	Description	Extraction Method and Moodle Nomenclature	Additional Information
Relevant actions	Number of relevant actions in the LMS	Total of relevant actions considered	Actions like log in, log out, profile updating, check calendar, refresh content, etc. are dismissed
Activity days	Number of days between the first and the last relevant action on every unit, for example, first action *check quiz* and last action 5 days later *quiz close attempt*	Date of last relevant action—date of first relevant action (in days)	Students have a period of 15 days to complete the unit

the module *assignment* and the variables related to the *actions*), from the date it was possible to view the task (Moodle also records this automatically from the module *assignment* and the variables related to the *views*).

Other reasonable variables are easily extracted through similar procedures. For example, variables such as *time spent on theoretical contents* or *time spent in forums* are not as reliable as we would like because the experience took place outside of teaching hours; thus, while using Moodle, the students could simultaneously be working or surfing the Internet. However, the variable *time spent in tasks* is a reliable indicator for this course because the Moodle module quiz allows a time limit to be set for every task. Here, we see some of the added difficulties of being out of the laboratory and in a real educational condition. In this regard, the time variables by themselves are very tricky. It might seem that the more time those students spend studying, the better grades they should receive, but the relationship is not as simple as this; it mainly depends on the quality of the studying time. For that reason, the value of these variables is necessarily linked to other relevant variables in the learning progress, such as the groups used in the automatic clustering.

Other examples of feasible indicators of the students' performance are the variables *typing time* and *number of words in* forums; the latter was selected for this tutorial. Based on common sense, the variable *typing time* could be mediated by a student's individual skills. Nevertheless, according to the literature, variables such as

number of messages sent to the forum or *number of forum messages read* are related to student achievement. Accordingly, we think that the mean *number of words* and *sentences* in posts would be a good indicator of the quality of the answers because students are asked to post a reflection. On that basis, we have chosen what we think are the most representative and objective variables available in the Moodle logs.

Next, it is necessary to transform or convert all this information (from Tables 1.4, 1.5, 1.6, and 1.7) into an attribute-relation file format (ARFF) summary file. This is the data format used by Weka (Witten et al., 2011), which is the DM tool used for the study's clustering. Weka is a collection of machine learning algorithms for DM tasks. The Weka system has several clustering algorithms; we used the expectation–maximization (EM) clustering algorithm. This algorithm is used in statistics to find maximum likelihood estimators of parameters in probabilistic models that rely on unobservable variables. We have selected this specific algorithm because it is a well-known clustering algorithm that does not require the user to specify the number of clusters. Our objective is to group together students who have similar characteristics when using Moodle. In order to do this, we used Weka Explorer (see Fig. 1.6): in the "Preprocess" tab, we clicked on the "Open file…" button and selected the previous summary .ARFF file. Then, we clicked on the "Cluster" tab and, in the "Clusterer" box, selected the "Choose" button. In the pull-down menu, we selected the cluster scheme "EM" and then clicked on the "Start" button to execute the algorithm.

When the training set was complete, the "Cluster" output area on the right panel of the "Cluster" window was filled with text describing the results of the

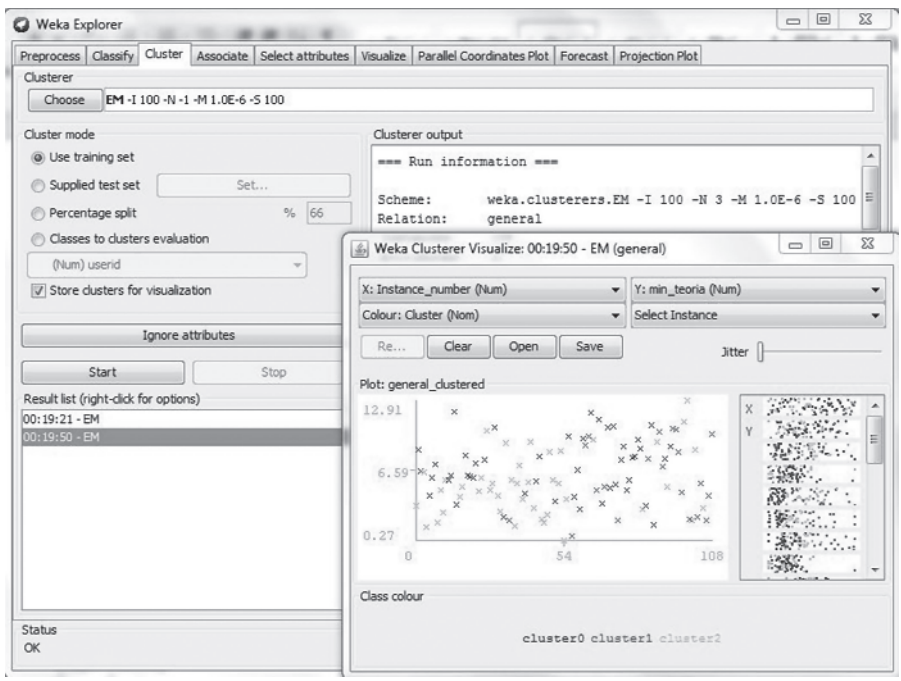

Figure 1.6 Weka clustering interface.

TABLE 1.8 Values (mean ± std.dev.) of the centroids of each cluster

Attribute	Cluster 0	Cluster 1	Cluster 2
Time theory	5.9 ± 3.2	7.1 ± 2.6	3.9 ± 1.5
Time tasks	14.1 ± 2.7	11.3 ± 6.2	5.3 ± 1.9
Time forums	8.7 ± 7.0	12.4 ± 5.4	7.6 ± 4.5
Days theory	6.0 ± 2.4	1.6 ± 6.9	8.4 ± 2.6
Days tasks	3.5 ± 1.0	1.8 ± 0.8	6.8 ± 2.3
Days "hand in"	4.8 ± 0.9	3.0 ± 1.2	9.3 ± 2.2
Number of words in forums	7.5 ± 6.5	9.2 ± 3.5	5.3 ± 3.4
Number of sentences in forums	92.9 ± 23.1	107.8 ± 40.6	78.1 ± 39.4

training and testing. In our case, we obtained three clusters, with the following distribution of students:

1. *Cluster 0*: 23 students (22 pass and 1 fail)
2. *Cluster 1*: 41 students (39 pass and 2 fail)
3. *Cluster 2*: 20 students (13 fail and 7 pass)

Clustering algorithms provide a highly interpretable result model by means of the values of each cluster centroid (Table 1.8). The centroid represents the most typical case/student or prototype in a cluster, and it does not necessarily describe any given case in that cluster.

As shown by the mean values in the different variables, students in Clusters 0 and 1 (in which most students passed) obtained higher values than those in Cluster 2 (in which most students failed) in terms of times (for theory, task, and forums) and counts (of words and sentences in forums). However, Clusters 0 and 1 had lower values for days (related to theory, tasks, and assignments hand in). Cluster 0 gives priority to the procedural level of knowledge, corresponding to the scores of the time and days tasks. The students comprising that cluster also seemed to show an achievement or strategic approach based on the prioritization of actions related to the compulsory assignments. In contrast, students belonging to Cluster 1 were presumably adopting a more dedicated approach to learning; note that the scores were good whether the variables were related to compulsory or suggested assignments. Finally, Cluster 2, comprised of students who normally fail, shows maladaptive variable levels and less frequent activity in the LMS.

1.3 WORKING WITH ProM

As is well known, instructors can easily gain insight into the way students work and learn in traditional learning settings. However, in LMS, it is more difficult for teachers to see how the students behave and learn in the system and to compare that system to other systems with structured interactions. These environments provide data on the interaction at a very low level. Because learner activities are crucial for an effective online teaching–learning process, it is necessary to search for empirical and effective tools to better observe patterns in the online environment; EPM and particularly ProM could be good resources for this purpose. Furthermore, the creation and evaluation of the models generated with ProM allow the researcher or instructor to not

only know more about the learning results but also to go through the learning process to better understand the referred results. Therefore, we used ProM (Van der Aalst, 2011a) for EPM. ProM is an extensible framework that supports a wide variety of PM techniques in the form of plug-ins.

Among the wide variety of algorithms, we applied the robust algorithm Heuristic Miner (Weijters, van der Aalst, & de Medeiros, 2006) to investigate the processes in the users' behavior. In this context, Heuristic Miner can be used to express the main behavior registered in an event log. It focuses on the control-flow perspective and generates a process model in the form of a Heuristics Net for the given event log. Therefore, the Heuristic Miner algorithm was designed to make use of a frequency-based metric that is less sensitive to noise and the incompleteness of the logs. As quality measures, we used fitness and the default threshold parameters of the Heuristic Miner algorithm. We applied Heuristic Miner using the ProM tool over the six previously obtained log data sets in order to discover students' process models and workflows. We applied the algorithm to each of these logs:

1. All students (84 students)
2. Students who passed (68 students)
3. Students who failed (16 students)
4. Students assigned to Cluster 0 (22 pass and 1 fail)
5. Students assigned to Cluster 1 (39 pass and 2 fail)
6. Students assigned to Cluster 2 (13 fail and 7 pass)

The first task after starting ProM was to import a log file in the following way: Click the "import…" icon in the upper-right corner and select the appropriate MXML file. The result is shown in Figure 1.7.

Figure 1.7 ProM interface for importing a log file.

Next, we could apply all kinds of ProM plug-ins. We could access all the available plug-ins by clicking in the ► tab in the upper middle bar (see Fig. 1.8).

From the list of plug-ins available in ProM, we selected "Mine for a Heuristic Net using Heuristics Miner" and click the "Start" button. Then, the parameters of Heuristic Miner were shown (see Fig. 1.9). The default values of these parameters were used in all our experiments.

Figure 1.8 List of plug-ins available in ProM.

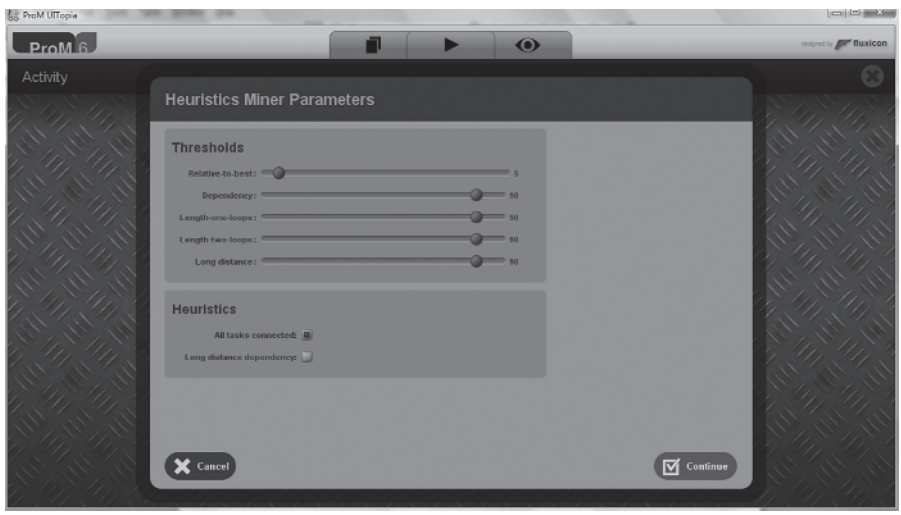

Figure 1.9 Parameters of the Heuristics Miner.

Once we pressed the "Continue" button on the configuration parameters screen, the discovered models are shown.

1.3.1 Discovered Models

The model discovered by the Heuristics Miner algorithm (Van der Aalst, 2011b) is a heuristic network that is a cyclic, directed graph representing the most common behaviors of students browsing the course. In this graph, the square boxes represent the actions of the students when interacting with Moodle's interface, and the arcs/links represent dependences/relations between actions. Next, we describe the discovered models using each of our log files.

Figure 1.10 shows the heuristic network obtained when using the log file with all students. We can see that there are two subnets that most of the students follow in the course. The upper subnet consists of *view forum* actions about the most viewed forums in the course, and the lower subnet consists of *view quiz* actions about the most viewed quizzes in the course. From the expert point of view, this information, although useful, is only a surface-level approach to the learning process, which we want to explore more deeply. It is important to note that these networks show the general behavior of all the students (fail and pass students mixed); probably for this reason, there are a lot of relations/dependences between the actions that make the model harder to interpret.

Figure 1.11 shows the heuristic network obtained when using only the logs of the *passing students*. We can see that this type of student followed a relatively high number of subnets. Respectively, these students followed seven subnets: *quiz view*, *quiz view summary*, *quiz attempt*, *quiz close attempt*, *quiz continue attempt*, *quiz review*, and *forum view*. Thus, we can see that passing students were very active in quiz actions. This makes sense, as based on the instructor's directions, success in the course was oriented to practical tasks. This model provides better insight into the students' learning processes than the previous model did. The students who passed the course performed well in the core subprocesses of learning: collaborative learning (*forum view*), forethought (*quiz view, quiz view summary*), performance (*quiz attempt*, *quiz close attempt, quiz continue attempt*), and self-reflection (*quiz review*).

Figure 1.12 shows the heuristic network obtained when using only the logs of *failing students*. This figure shows the two subnets used by most of the students who failed the course. The top subnet consists of *page view* actions for the most viewed pages in the course's text content. The bottom subnet consists of *view quiz* actions for the most viewed quizzes. Taking into account that the actions related to practical tasks and forums are not especially notable, we could conclude that, rather than engaging in the task and talk in the forums, the failing students could be using their time to study the theoretical contents. However, as previously mentioned, the course's goal was not the acquisition of declarative knowledge but putting this knowledge into practice. Based on this simple fact, these students' learning is incorrectly oriented.

It is also interesting to see how the heuristic net of students who failed is much smaller than those of the heuristic net for all students and for passing students. On the

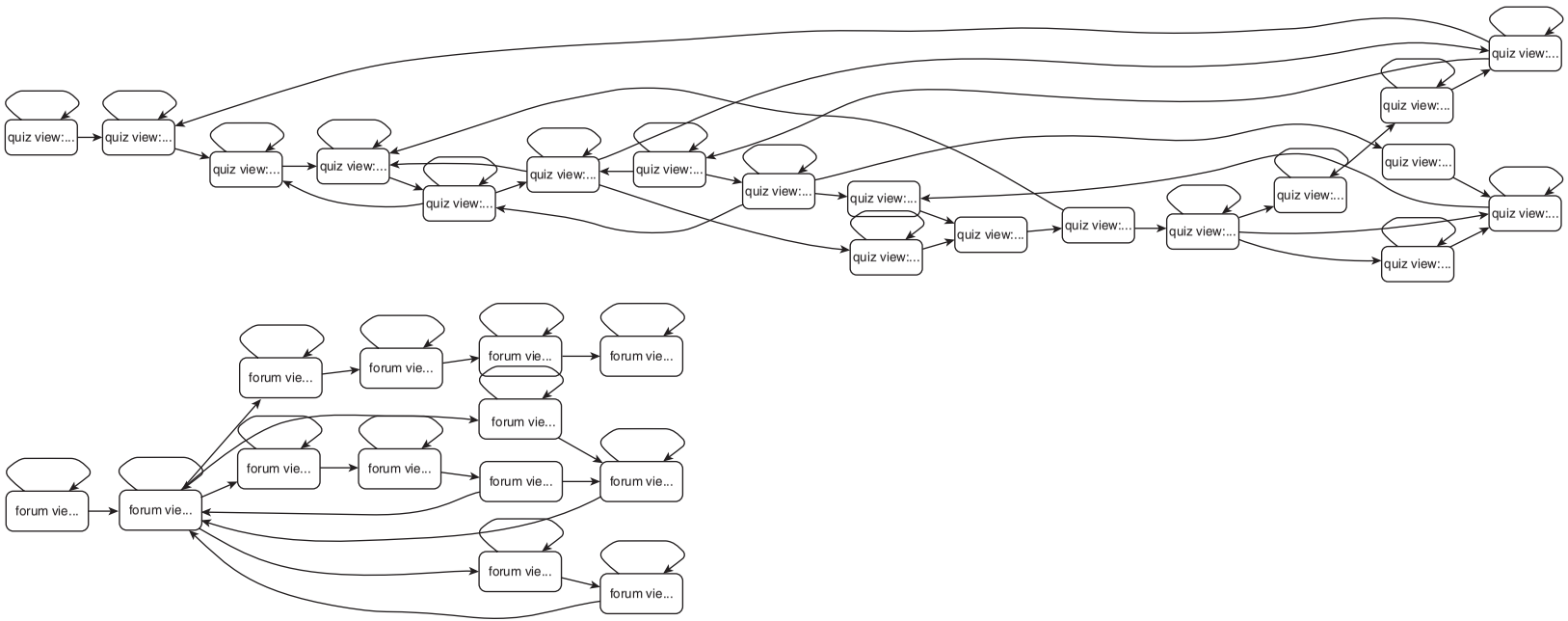

Figure 1.10 Heuristic net of all students.

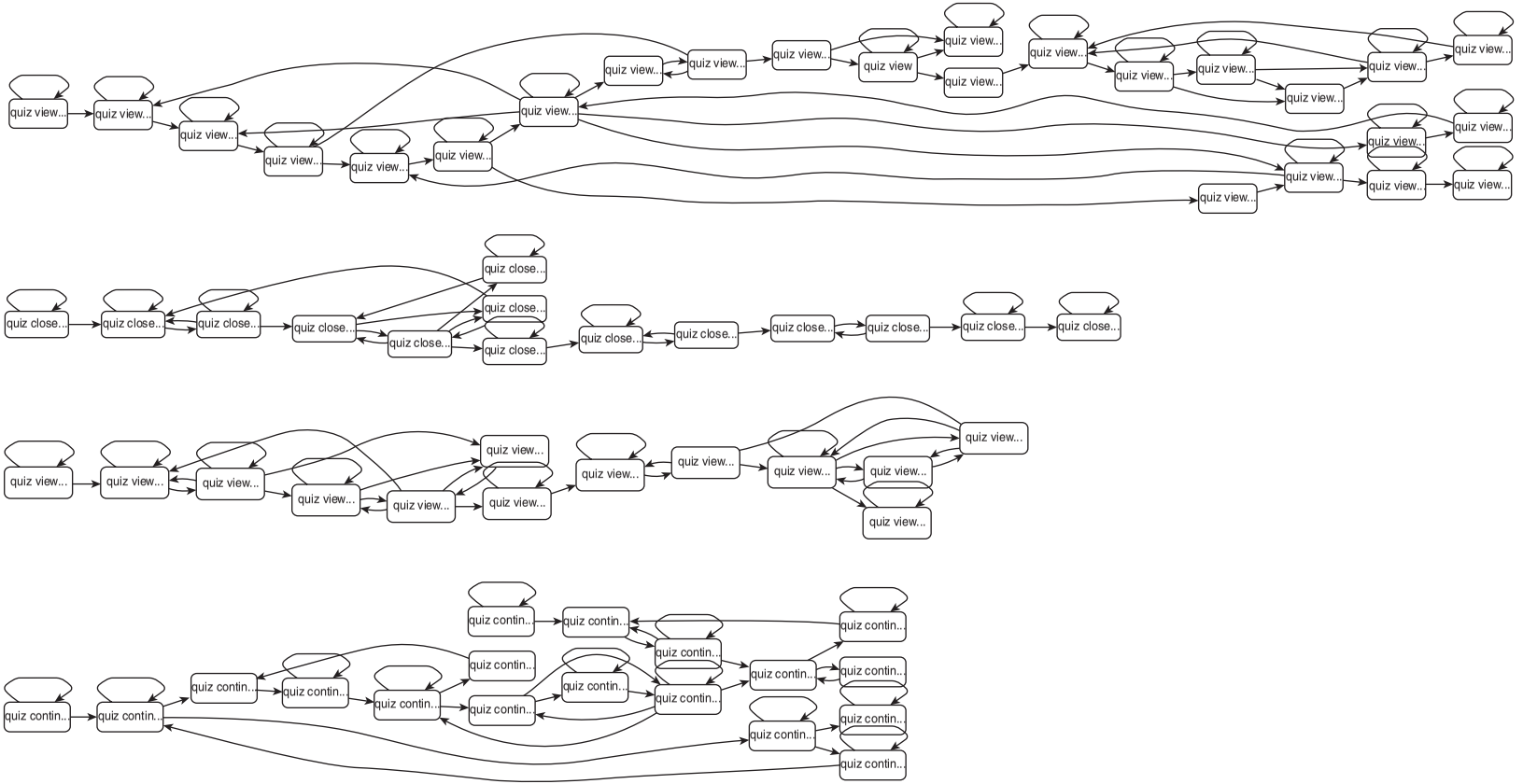

Figure 1.11 Heuristic net of passing students.

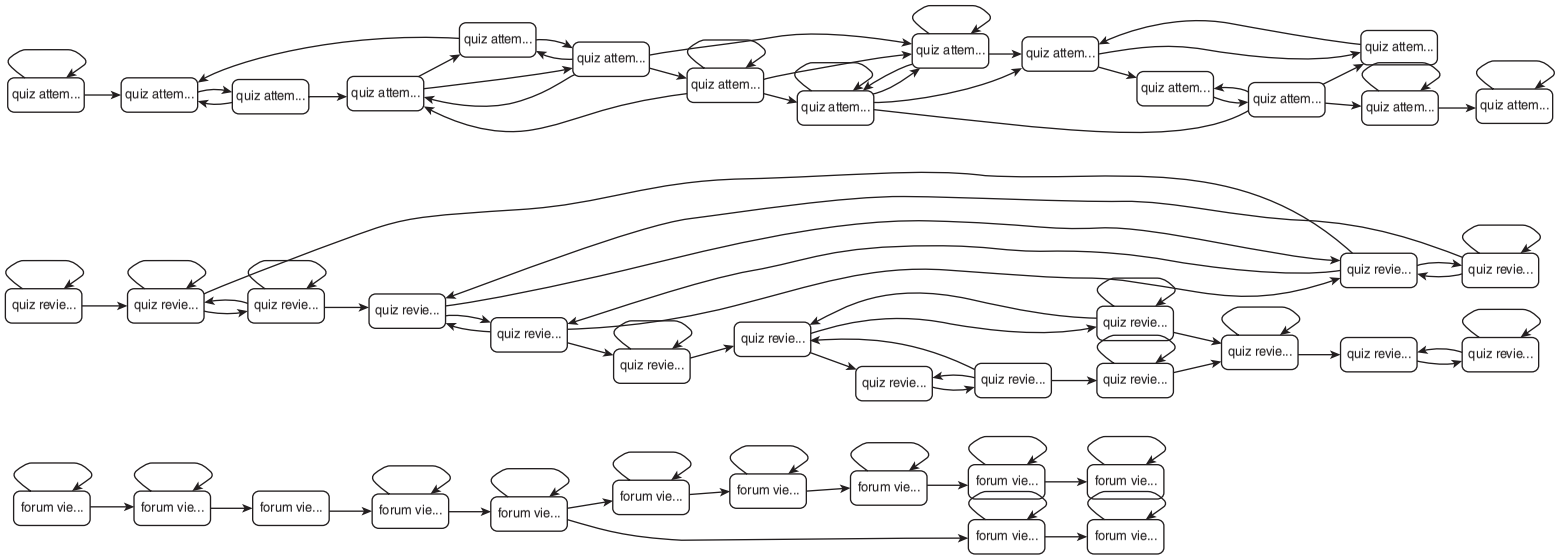

Figure 1.11 (*Continued*)

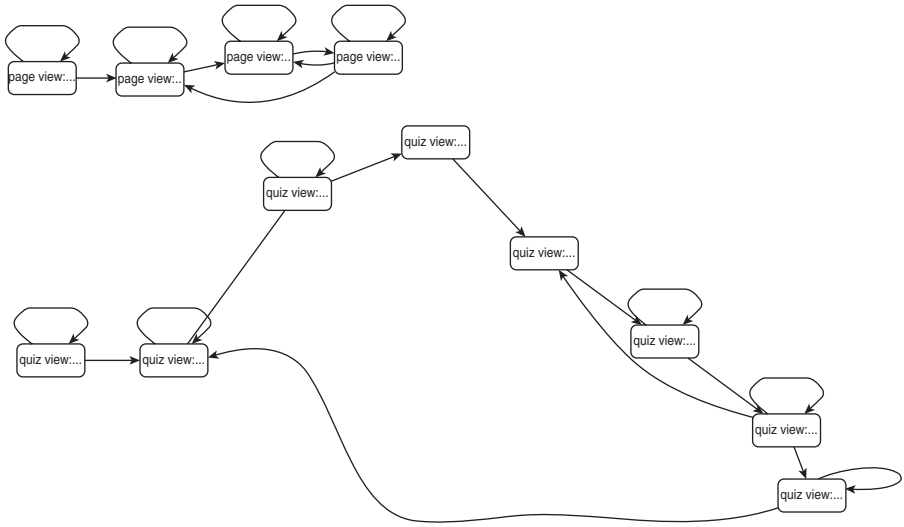

Figure 1.12 Heuristic net of failing students.

other hand, this chapter has already offered warnings about wrong assumptions related to study time. This could also be applied to the number of actions; having more interactions with the LMS does not necessarily reflect better performance. However, the evident scarcity of interaction with the learning environment showed by failing students in this particular case is conclusive. In the end, based on this chapter's scope, there is good news, as the behavior of failing students would be easy to detect and interpret.

Finally, we show one example of the three clusters. Figure 1.13 shows the heuristic net obtained when using only the logs of passing students in Cluster 0. This figure shows the five subnets followed by this subtype of students. According to this model of passing students from Cluster 0, the obtained subnets consist of these actions: *quiz view*, *quiz close attempt*, *quiz review*, *quiz continue attempt*, and *forum view*. If we compare Figure 1.11 (all students who pass) with Figure 1.13 (a subtype of students who pass), we can see that not only are there fewer subnets (five instead of seven) but also that the networks are smaller (with a smaller number of nodes and arcs). The usefulness of this, apart from generating models that are easier to interpret, is that the instructor of the course knows and can select which are the crucial variables for successful performance in the course and/or which target variables will determine the ProM model generation from the previous clustering.

1.3.2 Analysis of the Models' Performance

We have also carried out an analysis of the performance of the previously obtained models (heuristic networks). In order to do this, a fitness measure is normally used (Ayutaya et al., 2012). Fitness is a quality measure indicating the gap between the

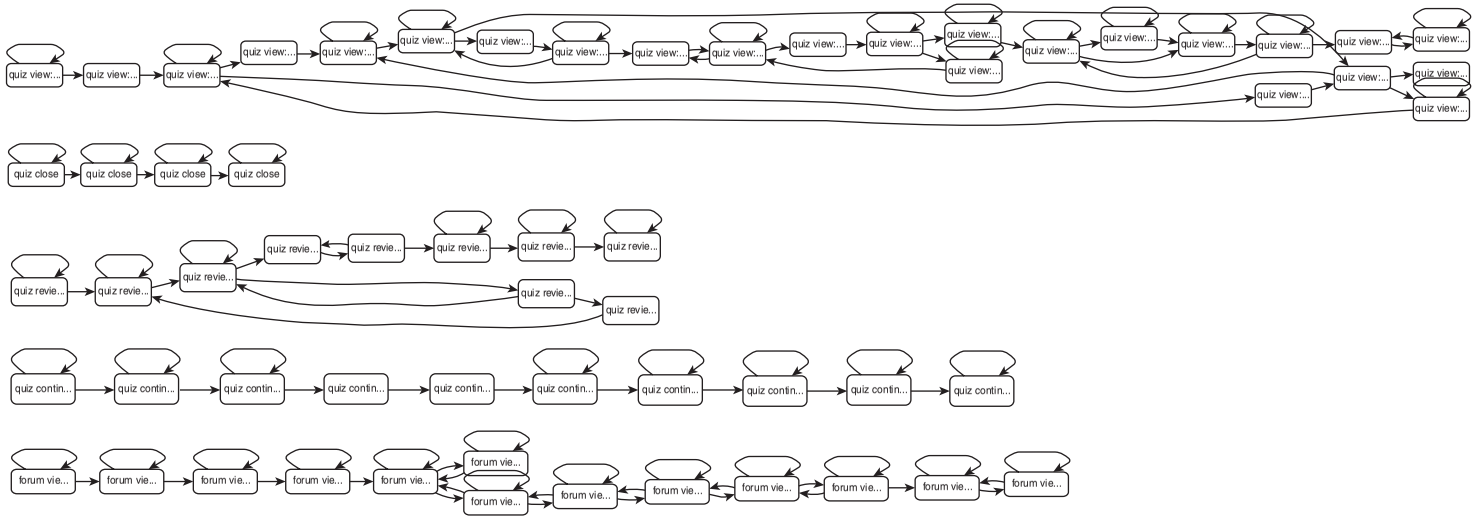

Figure 1.13 Heuristic net of Cluster 0 students.

behavior actually observed in the log and the behavior described by the process model. It gives the extent to which the log traces can be associated with the execution paths specified by the process model. If a model has a poor fitness value, this indicates that the mined process model does not successfully parse most of the log traces. The fitness results that we have obtained for each of our models are shown in Table 1.9.

As we can see in Table 1.9, the lowest fitness was obtained when using all data, for which 70 of 84 students fit to the obtained model (83.33%). On the other hand, all the other models (using both manual and automatic clustering) obtained a fitness value greater than 90% in all cases. The highest fitness value was obtained when using data from students who failed, for which 15 of 16 students fit the obtained model (93.75%). Thus, in this case, we can see that these specific models, which were obtained using manual and automatic grouping/clustering, performed better than the general model obtained from all students.

Additionally, we evaluated the compressibility of the obtained models. In order to do this, the complexity or size of each model is normally used (Bogarín et al., 2014). We have used two typical measures from graph theory: the total number of nodes and the total number of links in the models/graphs (see Table 1.10).

As we can see in Table 1.10, the smallest (most comprehensible) model was obtained with the data set of failing students, followed by those for all students and Cluster 2 students. The other three obtained models were much larger and more complex, especially the model using passing students.

TABLE 1.9 Fitness of the obtained models

Data Set	Fitness
All students	0.8333
Pass students	0.9117
Fail students	0.9375
Cluster 0 students	0.9130
Cluster 1 students	0.9024
Cluster 2 students	0.9000

TABLE 1.10 Complexity/size of the obtained models

Data Set	Number of Nodes	Number of Links
All students	32	70
Pass students	113	244
Fail students	12	24
Cluster 0 students	61	121
Cluster 1 students	59	110
Cluster 2 students	38	84

1.4 CONCLUSION

The present work proposed using clustering to improve EPM and, at the same time, optimize both the performance/fitness and comprehensibility/size of the model. We have obtained different models by using data sets from different groups of students:

- In the model from the data set of all students, the students showed different behavior and only had a few common actions because there was a mix of different types of students (*passing* and *failing*).
- In the model from the data set of students who failed in Cluster 2 students, the students only showed a few common behavioral patterns because these types of students (who failed) were less participatory or interactive when browsing the course than were the others (who passed).
- In the model from the data set for students who passed in Cluster 0 and Cluster 1, the students showed a much higher number of common behavioral patterns because these types of students (who passed) were more active users than the others (who failed).

From an educational and practical point of view—to be able to use this information for providing feedback to instructors about student learning—these models could easily be used to point out which new students are at risk of failing a course. For example, instructors only have to check to see which new students are following the same routes/behavioral patterns shown by the heuristic net of students who failed.

On the other hand, model comprehensibility is a core goal in education due to the transfer of knowledge that this entails. Making graphs, models, and visual representations understandable to teachers and students makes these results essential for monitoring the learning process and providing feedback; one of our goals is to do precisely that in real time. Furthermore, Moodle does not provide specific visualization tools for students' usage data that would allow the different agents involved in the learning process to understand the large amount of raw data and to become aware of what is happening in distance learning. In addition the results can also be extended to the improvement of adaptive hypermedia learning environments, for which prompting the students about recommended learning paths, shortcuts, etc. is the basis for enhancing the learning experience in a more strategic way.

Finally, in the near future, more experiments will be conducted to test our approach using other types of courses from different fields of knowledge. We also want to explore other ways to group students before PM. For example, we could group students based on the triangulation of different sources of information, such as self-reported data or psychophysiological measures based on students' metacognitive behavior. We also want to test if clustering the content (or even a manual semantic mapping of the content/course structure) would allow us to simplify the process models. Even further, we could split the course into semantic blocks and see how students progress—either as a logical progression (e.g., unit 1.8) or as a time sequence (e.g., week 1–10). This would be a very interesting way to identify faults in the process. For example, students progressing through units in similar ways are more likely to perform better (i.e., they have a strategy).

ACKNOWLEDGMENTS

This research is supported by projects of the Spanish Ministry of Science and Technology TIN2014-55252-P, EDU2014-57571-P, and the European Funds for Regional Development Ref. GRUPIN14-053.

REFERENCES

Ayutaya, N. S. N., Palungsuntikul, P., & Premchaiswadi, W. Heuristic mining: Adaptive process simplification in education. International Conference on ICT and Knowledge Engineering, 221–227, November 21–23, 2012.

Azevedo, R., Behnagh, R., Duffy, M., Harley, J. M., & Trevors G. J. Metacognition and self-regulated learning in student-centered learning environments. In Jonassen, D. H. & Land, S. M. (eds), *Theoretical Foundations of Student-Center Learning Environments*, 2nd edition. Erlbaum, Mahwah, NJ, 216–260, 2012.

Bannert, M., Reimann, P., & Sonnenberg, C. Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and Learning*, 9(2), 161–185, 2014.

Bogarín, A., Romero, C., Cerezo, R., & Sánchez-Santillán, M. Clustering for improving educational process mining. International Conference on Learning Analytics and Knowledge, New York, 11–15, 2014.

Boiarsky, C. A model for analyzing revision. *Journal of Advanced Composition*, 5, 65–78, 1984.

Cairns, A. H., Ondo, J. A., Gueni, B., Fhima, M., Schwarcfeld, M., Joubert, C., & Khelifa, N. Using semantic lifting for improving educational process models discovery and analysis. *CEUR Workshop Proceedings*, 1293, 150–161, 2004.

Cairns, A. H., Gueni, B., Fhima, M., Cairns, A., David, S., & Khelifa, N. Towards custom-designed professional training contents and curriculums through educational process mining. IMMM 2014, The Fourth International Conference on Advances in Information Mining and Management, 53–58, 2014.

Dutt, A., Aghabozrgi, S., Ismail, M. A. B., & Mahroeian, H. Clustering algorithms applied in educational data mining. *International Journal of Information and Electronics Engineering*, 5, 112–116, 2015.

Romero, C., Ventura, S., & García, E. Data mining in course management systems: Moodle case study and tutorial. *Computers & Education*, 51(1), 368–384, 2008.

Schoor, C. & Bannert, M. Exploring regulatory processes during a computer-supported collaborative learning task using process mining. *Computers in Human Behavior*, 28(4), 1321–1331, 2012.

Sedrakyan, G., Snoeck, M., & De Weerdt, J. Process mining analysis of conceptual modeling behavior of novices–empirical study using JMermaid modeling and experimental logging environment. *Computers in Human Behavior*, 41, 486–503, 2014.

Southavilay, V., Yacef, K., & Calvo, R. A. Process mining to support student's collaborative writing. Educational Data Mining Conference, 257–266, 2010.

Trcka, N. & Pechenizkiy, M. From local patterns to global models: Towards domain driven educational process mining. International Conference on Intelligent Systems Design and Applications, Milan, Italy, 1114–1119, 2009.

Van der Aalst, W. M. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer, Berlin/Heidelberg/New York, 2011a.

Van der Aalst, W. M. Using process mining to bridge the gap between BI and BPM. *IEEE Computer*, 44(12), 77–80, 2011b.

Van der Aalst, W. M., Adriansyah, A., de Medeiros, A. A., Arcieri, F., Baier, T., Blickle, T., & Pontieri, L. Process mining manifesto. Business Process Management Workshops, 169–194, 2012.

Vellido, A., Castro, F., & Nebot, A. Clustering educational data. In Romero, C., Ventura, S., Pechenizkiy, M., & Baker, R. (eds), *Handbook of Educational Data Mining*. CRC Press, Boca Raton, FL, 75–92, 2011.

Weijters, A. J. M. M., van der Aalst, W. M., & de Medeiros, A. A. Process mining with the Heuristics Miner-algorithm. Technische Universiteit Eindhoven, Tech. Rep. WP, 166, 2006.

Witten, I. H., Frank, E., & Hall, M. A. *Data Mining, Practical Machine Learning Tools and Techniques*, 3rd edition. Morgan Kaufman Publishers, Amsterdam, 2011.