Analytics on Moodle Data Using R Package for Enhanced Learning Management

V. Rachel¹#, Dr. G. Sudhamathy², Dr. M. Parthasarathy³

¹Research Scholar, Department of Educational Technology, Bharathiar University, Coimbatore, India.
²Assistant Professor, Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India.
³Assistant Professor, Department of Educational Technology, Bharathiar University, Coimbatore, India.

Abstract

There are a lot of learning analytics tools being developed which is integrated into the Moodle Course Management System for analysing the student’s performance and behaviour for enhanced learning management. But, most of these just concentrate on one technique such as data mining or statistical analysis or visualization. Also, they do not explore the capabilities of upcoming open source data analytics tool, the R Package which can perform complete data analytics applying the statistical techniques, data mining techniques and can produce effective visualization of the results. Hence, in this paper we are attempting to explore the data analytics capabilities of R Package and apply the same on a set of Moodle data which is captured on the learning behaviour of computer science students of a higher education institute. This analytics package can be integrated with Moodle, so that it can fetch the log files produced from Moodle continuously and produce the results of learning trends of the students.

Keywords: Data Analytics, Moodle, Learning Analytics, Educational Data Mining, R Package

I. INTRODUCTION

One of the most popular open source Course Management Systems (CMS) is the Moodle (Modular Object-Oriented Dynamic Learning Environment). Moodle is an open source software distributed under the GNU General Public License and it is written in the PHP programming language. Moodle was created to help educators to create online courses by Martin Dougiamas. Moodle is in continual evolution and it focuses on interaction and collaborative construction of the content. Moodle is designed in a modular fashion and it is helpful in creating new courses by adding content to it. Moodle has the components that consist of the activities and resources, such as forums, messaging, tests, assignments, wiki, blogs, glossaries, etc.

Moodle logs all the information about the students’ interactions in it. This log data, gives the information of how and when students perform their visits, quizzes, assignments, forum interactions, course views etc. Even though Moodle's log data can be filters using the reporting feature available in it, the resultant information is not adequate to provide meaningful information about the course and the student's performance. Hence, it is required to extract this log data and transform it into useful actionable knowledge and this is a difficult task. There are many methodologies and techniques available in this emerging field such as Educational Data Mining (EDM) and Learning Analytics (LA).

There are many such tools that aim to facilitate the process of discovering knowledge from educational data. There are many tools that extract the educational data and analyses it to produce valuable knowledge. Some of them perform ad-hoc analysis for a specific issue and some are developed for a more general purpose. Of these lists of tools only few consider Moodle data and produce valuable insights. Many of the existing tools only provide certain level of basic analytics and graphs about the user interaction patterns and they do not use any of the Data Mining techniques. Even though the existing tools are useful to get a glimpse of the course status, they are unable to discover new, interesting, and useful knowledge.

Hence, to overcome this problem we have developed an analytics framework that uses the techniques available in the R Package, an open source software distributed under the GNU General Public License. This framework does the complete list of tasks of Moodle log data extraction, pre-processing, data transformation, statistical analysis, clustering, classification, association rule mining and presenting the findings in the form of visual graphs and tables.

II. REVIEW OF LITERATURE

There are a lot of EDM (Educational Data Mining) and LA (Learning Analytics) tools that have been developed to solve the educational problems. New tools are being developed everyday to address different educational needs and objectives. Even though there are many tools that use data mining or analytics in the area of education, there are only few tools that is based on Moodle data. CoSyLMSAnalytics[20], is a web based tool based on Moodle data and this tool allows the teacher to assess and monitor the
progress of each student and provide feedback. This tool extracts the students access patterns by using correlation analysis and association rule mining techniques. ViMoodle[18], uses the Moodle log data to understand the students activities, quiz, forums, resources, etc., and it returns social network and snail graphs that uses the clustering and association rule mining techniques.

The tool CIECoF[17], provides recommendations on how to improve Moodle courses by using association rule mining and collaborative filtering techniques. Meerkat-ED tool [15], enables student’s participation in Moodle discussion forums and this data is analyzed using social network analysis techniques. MMT Tool [14], executes all the steps in the data mining process that applies pre-processing, classification, and association rule mining algorithms over data files created from Moodle database. The tool DRAL[9], reveals activities of learners of Moodle based on features extracted from log data. It uses techniques like grammar guided genetic programming and multi-objective for classification of the tasks. Most of the new tools on EDM and LA are web-based. E-learningWebMiner[11], reveals student’s behaviour profiles that instructors may use to improve their courses. It provides visualization graphs and uses clustering and association rule mining techniques. CVLA[1], integrates analytics techniques with the Moodle report that uses multiple data sets and analytics techniques in a single interface that uses network analysis and classification algorithms for predicting assignment submission.

IntelliBoard.net[3], extracts the statistical data available in Moodle and presents it on a dashboard in the form of printable charts, graphs, and multiple-format reports. SmartKlass[6], measures and analyses the learning process using an Open source and multi-platform learning analytics dashboard plug-in. MEAP[2], provides feedback on the level of engagement of a student on a Moodle course and it identifies activities, that have an impact on student success. Analytics Graphs[4], graphically summarises students access profile by providing three graphs that may facilitate pedagogical decisions. VeLA[5], is a framework that provides different representations of the information from different LMSs such as Moodle and displays it interactively so that the teachers can understand the students learning process.

Thus, we can see from the Table 1, that many of the existing tools that use the Moodle data are developed using PHP and Java and they are mainly meant to provide visualization results only. None of the existing tools explore the use of the open source software, R Package which is now extensively used for data analytics and data science in most of the fields. Also, no predominant pre-processing and data transformation techniques are applied so far. There is no tool that has the integrated features of data pre-processing, data transformation, statistical analysis, clustering analysis, classification analysis and association rule mining analysis with effective visualization. Hence, in this work we provide a complete analytics framework that integrates all the above said techniques and algorithms.

![Table 1. Comparison of Features of Moodle based Tools](image-url)
III. METHODOLOGY

The step by step approach of the proposed methodology is explained in the Fig. 1.

### Data Loading

The log files taken from the Moodle Tool installed in the Computer Science Department of an Institution is loaded as comma separated value file in the R Studio. The unloaded dataset under study (“logs_20170523-1240.csv”) contains the log information of the Moodle usage of two sections of students in the department for enhanced learning. The students use this tool during their course period for enhancing their performance in a particular subject. This file consists of 33811 records and it has a header record that has the fields as listed in Table 2.

### Data Pre-Processing

The log file loaded into R-Studio and the records are pre-processed, so that only relevant and correct information will be processed further. The records with no user name and those records with user names "Admin ADU" and "Guest User" are removed. Log data from the client are removed as only log data from the server are considered. Log data produced by the System component are not considered and hence they are removed. Creating the Time Stamp field, parsing and extracting the required fields for further processing (Timestamp, Username, Component, Event, IP address) are also done in the pre-processing step. Finally, duplicate records are removed after parsing.

### Data Transformation

The Data Transformation Step consists of the below sub steps. We first apply the group functions to get the User Events, User Visits, User Quizzes, User Assignments, User Forums Created, User Forums Read, User Course Views, User Files Submitted and User Grade Views. Then, we merge the grouped columns using Joins to obtain the entire transformed summary dataset. The grouping and joining of data are done using the sqldf package in R Studio which helps to coin complex database queries in the R Environment. After this, we split the transformed data into two separate datasets, which consists of the summary information of the two sections / classes (Class 1 & Class 2) involved in the study. The lists of attributes after transformation are as listed in the Table 3.

### Data Exploration

The Data Exploration Step consists of the below sub steps. The results of this step are obtained by using the sqldf package and the lattice package in R Studio which helps in

---

**Table 2. Attributes in the Log File**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>Time Stamp of the Log Record</td>
</tr>
<tr>
<td>User full name</td>
<td>Users Full Name</td>
</tr>
<tr>
<td>Affected user</td>
<td>Affected Users Full Name</td>
</tr>
<tr>
<td>Event context</td>
<td>Context of the Event</td>
</tr>
<tr>
<td>Component</td>
<td>Component Producing the Log Record</td>
</tr>
<tr>
<td>Event name</td>
<td>Name of the Event</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the Event</td>
</tr>
<tr>
<td>Origin</td>
<td>Origin of the Log Record (Client / Web Server)</td>
</tr>
<tr>
<td>IP address</td>
<td>IP Address of the Device through Which the User Logged in</td>
</tr>
</tbody>
</table>

**Table 3. Attributes after Data Transformation**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Name of the Student</td>
</tr>
<tr>
<td>Visits</td>
<td>Total No. of Visits made by the Student</td>
</tr>
<tr>
<td>Quizzes</td>
<td>Total No. of Quizzes taken by the Student</td>
</tr>
<tr>
<td>Assignment</td>
<td>Total No. of Assignments by the Student</td>
</tr>
<tr>
<td>ForumCreated</td>
<td>Total No. of Forums Created by the Student</td>
</tr>
<tr>
<td>ForumRead</td>
<td>Total No. of Forum Reads by the Student</td>
</tr>
<tr>
<td>CourseViews</td>
<td>Total No. of Course Views by the Student</td>
</tr>
<tr>
<td>FileSubmission</td>
<td>Total No. of File Submissions by the Student</td>
</tr>
<tr>
<td>GradeViews</td>
<td>Total No. of Grade Views by the Student</td>
</tr>
</tbody>
</table>
querying and extracting the required data and presenting them as bar-charts. The functions used are `sqlf()` and `barchart()`.

1) Categorizing Visits into Low, Medium and High for Class 1
2) Categorizing Visits into Low, Medium and High for Class 2
3) Visualizing Visit Frequency of Class 1
4) Visualizing Visit Frequency of Class 2
5) Visualizing Quiz Frequency of Class 1
6) Visualizing Quiz Frequency of Class 2
7) Visualizing Assignment Frequency of Class 1
8) Visualizing Assignment Frequency of Class 2
9) Visualizing Forums Created Frequency of Class 1
10) Visualizing Forums Created Frequency of Class 2
11) Visualizing Forums Read Frequency of Class 1
12) Visualizing Forums Read Frequency of Class 2
13) Visualizing Courses Viewed Frequency of Class 1
14) Visualizing Courses Viewed Frequency of Class 2
15) Visualizing File Submissions Frequency of Class 1
16) Visualizing File Submissions Frequency of Class 2
17) Visualizing Grade Views Frequency of Class 1
18) Visualizing Grade Views Frequency of Class 2

Statistical Analysis of Data

This step is meant for analyzing the statistical properties of the datasets which would help in further application of data mining techniques to these datasets. We get the summary statistics of Class 1 and Class 2 datasets separately. This helps in understanding the attributes range of values, its mean, median values etc. Then we obtain the correlation matrix plot of the attributes of the Class 1 and Class 2 datasets. This helps to identify which of the attributes are correlated, so that we can understand the influence of one factor on the other. Finally, we obtain the scatter plot of the correlated attributes in each of these datasets to visualize the correlation pattern. The R functions `summary()`, `corplot()` [from `corrplot` package] and `plot()` are used to obtain the results in this step.

Clustering Analysis of Data

In this step, as we have two separate datasets for each of the classes, we obtain the optimal number of clusters for each of these datasets and apply K-Means Clustering on them. We obtain the tabular list of students belonging to the various clusters and how the various parameters of learning are ranged in these clusters. The results of the clustering applied to the various parameters like Visits, Course Views, Quizzes and Assignments are also visualized as scatter plots. The function `pamk()` in the `fpc` package is used to obtain the optimal number of clusters from the datasets. The `kmeans()` function in the `stats` package is used for clustering the datasets.

Classification of Data

In this step, as classification is a supervised learning method, we first assign labels to the various parameters like Visits, Quizzes, Assignments, Course Views and File Submissions as “HIGH”, “MEDIUM” and “LOW” based on their range of values. For the parameter Grade, we assign the labels “EXCELLENT”, “GOOD” and “FAIL” based on its range of values. These labelling is done using the `sqlf()` function. Then we split the dataset containing both the section data into training and test datasets in the ratio of 80% and 20% respectively. We then, apply the decision tree classification technique using the `ctree()` function in the `party` package on the training dataset. The results of this classification can be seen as a visual decision tree. Then we also predict the classes of the test dataset using the `predict()` function.

Association Rule Mining of Data

In this step, we apply the Apriori Algorithm on the entire transformed dataset containing the two sections data and for which the labels are assigned for the various parameters. The `apriori()` function in the `arules` package in the R Studio is used to obtain the association rules. The rules produced have the three top measures of association rule mining, namely the support, confidence and lift. The produced rules are then ordered based on the measure lift in the descending order using the function `sort()`. Then redundant rules are pruned from the produced lot of rules using the `which()` function. Finally we inspect the rules using the `inspect()` function and visualize the pruned rules as group matrix using the `plot()` function in the `arulesViz` package.

IV. RESULTS AND DISCUSSIONS

The input Moodle log file “logs_20170523-1240.csv” is loaded into the R Studio, it is pre-processed, transformed using complex SQL queries and then further analysis is done on the transformed data. The total number of records before pre-processing is 33811 and the number of records after pre-processing is 5404. After data transformation is identified that the total number of students in Class 1 are 48 and in Class 2 are 46. The lists of attributes after transformation are as listed in the Table 3.

The results of the Data Exploration steps are shown in Fig. 2 to Fig. 17. The results of the Statistical Analysis of Data are shown in Table 4 and Table 5 and in Fig. 18 to Fig. 23. The results of the Clustering Analysis of Data are shown in Table 6 to Table 9 and in Fig. 24 to Fig. 29. The results of the
Classification of Data can be seen in Table 10 and Table 11 and in the Fig. 30. The results of the Association Rule Mining of Data can be seen in Table 12 and in the Fig. 31.

3) Visualizing Visit Frequency of Class 1

[Bar Chart for Visit Frequency of Class 1]

Fig. 2. Bar Chart for Visit Frequency of Class 1

4) Visualizing Visit Frequency of Class 2

[Bar Chart for Visit Frequency of Class 2]

Fig. 3. Bar Chart for Visit Frequency of Class 2
5) Visualizing Quiz Frequency of Class 1

![Fig. 4. Bar Chart for Quiz Frequency of Class 1 Students](image_url)

6) Visualizing Quiz Frequency of Class 2

![Fig. 5. Bar Chart for Quiz Frequency of Class 2 Students](image_url)
7) Visualizing Assignment Frequency of Class 1

![Bar Chart for Assignment Frequency of Class 1 Students](image1)

**Fig. 6.** Bar Chart for Assignment Frequency of Class 1 Students

8) Visualizing Assignment Frequency of Class 2

![Bar Chart for Assignment Frequency of Class 2 Students](image2)

**Fig. 7.** Bar Chart for Assignment Frequency of Class 2 Students
9) Visualizing Forums Created Frequency of Class 1

![Fig. 8. Bar Chart for Forums Created Frequency of Class 1 Students](image)

10) Visualizing Forums Created Frequency of Class 2

![Fig. 9. Bar Chart for Forums Created Frequency of Class 2 Students](image)
11) Visualizing Forums Read Frequency of Class 1

![Bar Chart for Forums Read Frequency of Class 1 Students](image1)

**Fig. 10.** Bar Chart for Forums Read Frequency of Class 1 Students

12) Visualizing Forums Read Frequency of Class 2

![Bar Chart for Forums Read Frequency of Class 2 Students](image2)

**Fig. 11.** Bar Chart for Forums Read Frequency of Class 2 Students
13) Visualizing Courses Viewed Frequency of Class 1

![Bar Chart for Frequency of Courses Viewed of Class 1 Students](image1)

Fig. 12. Bar Chart for Frequency of Courses Viewed of Class 1 Students

14) Visualizing Courses Viewed Frequency of Class 2

![Bar Chart for Frequency of Courses Viewed of Class 2 Students](image2)

Fig. 13. Bar Chart for Frequency of Courses Viewed of Class 2 Students
15) Visualizing File Submissions Frequency of Class 1

Fig. 14. Bar Chart for Frequency of File Submissions by Class 1 Students

16) Visualizing File Submissions Frequency of Class 2

Fig. 15. Bar Chart for Frequency of File Submissions by Class 2 Students
17) Visualizing Grade Views Frequency of Class 1

Fig. 16. Bar Chart for Frequency of Grade Views by Class 1 Students

18) Visualizing Grade Views Frequency of Class 2

Fig. 17. Bar Chart for Frequency of Grade Views by Class 2 Students
Table 4. Summary Statistics of Class 1 Log Data

<table>
<thead>
<tr>
<th></th>
<th>Visits</th>
<th>Quizzes</th>
<th>Assignment</th>
<th>ForumCreated</th>
<th>ForumRead</th>
<th>CourseViews</th>
<th>FileSubmission</th>
<th>GradeViews</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>35.00</td>
<td>1.000</td>
<td>3.000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>1.000</td>
<td>5.000</td>
<td>0.0000</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>50.75</td>
<td>2.000</td>
<td>4.000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>6.75</td>
<td>6.000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Median</td>
<td>65.50</td>
<td>2.000</td>
<td>4.000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>12.50</td>
<td>6.000</td>
<td>0.0000</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>79.00</td>
<td>3.000</td>
<td>5.000</td>
<td>0.00000</td>
<td>2.00000</td>
<td>22.25</td>
<td>7.250</td>
<td>1.0000</td>
</tr>
<tr>
<td>Max.</td>
<td>131.00</td>
<td>5.000</td>
<td>7.000</td>
<td>1.00000</td>
<td>5.00000</td>
<td>44.00</td>
<td>10.000</td>
<td>6.0000</td>
</tr>
</tbody>
</table>

Table 5. Summary Statistics of Class 2 Log Data

<table>
<thead>
<tr>
<th></th>
<th>Visits</th>
<th>Quizzes</th>
<th>Assignment</th>
<th>ForumCreated</th>
<th>ForumRead</th>
<th>CourseViews</th>
<th>FileSubmission</th>
<th>GradeViews</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>16.00</td>
<td>0.000</td>
<td>1.000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>1.000</td>
<td>2.000</td>
<td>0.0000</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>34.00</td>
<td>1.000</td>
<td>2.000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>2.000</td>
<td>4.000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Median</td>
<td>39.50</td>
<td>1.000</td>
<td>3.000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>3.500</td>
<td>6.000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Mean</td>
<td>44.70</td>
<td>1.239</td>
<td>2.978</td>
<td>0.02174</td>
<td>0.52174</td>
<td>5.652</td>
<td>6.174</td>
<td>0.2609</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>52.75</td>
<td>1.000</td>
<td>3.000</td>
<td>0.00000</td>
<td>1.00000</td>
<td>6.750</td>
<td>6.000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Max.</td>
<td>125.00</td>
<td>4.000</td>
<td>6.000</td>
<td>1.00000</td>
<td>4.00000</td>
<td>28.000</td>
<td>14.000</td>
<td>4.0000</td>
</tr>
</tbody>
</table>

**Correlation Analysis of Class 1**

![Correlation Matrix of Class 1 Log Data](image)

Fig. 18. Plot of Correlation Matrix of Class 1 Log Data
### Correlation Analysis of Class 2

<table>
<thead>
<tr>
<th></th>
<th>Visits</th>
<th>Quizzes</th>
<th>Assignment</th>
<th>ForumCreated</th>
<th>ForumRead</th>
<th>CourseViews</th>
<th>FileSubmission</th>
<th>GradeViews</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visits</td>
<td>1</td>
<td>0.34</td>
<td>0.63</td>
<td>0.34</td>
<td>0.79</td>
<td>0.85</td>
<td>0.71</td>
<td>0.39</td>
</tr>
<tr>
<td>Quizzes</td>
<td>0.34</td>
<td>1</td>
<td>-0.17</td>
<td>0.34</td>
<td>0.26</td>
<td>0.47</td>
<td>-0.13</td>
<td>0.12</td>
</tr>
<tr>
<td>Assignment</td>
<td>0.63</td>
<td>-0.17</td>
<td>1</td>
<td>0.4</td>
<td>0.43</td>
<td>0.24</td>
<td>0.96</td>
<td>0.21</td>
</tr>
<tr>
<td>ForumCreated</td>
<td>0.34</td>
<td>-0.05</td>
<td>0.4</td>
<td>1</td>
<td>0.56</td>
<td>0.2</td>
<td>0.36</td>
<td>0.48</td>
</tr>
<tr>
<td>ForumRead</td>
<td>0.79</td>
<td>0.26</td>
<td>0.43</td>
<td>0.56</td>
<td>1</td>
<td>0.75</td>
<td>0.45</td>
<td>0.58</td>
</tr>
<tr>
<td>CourseViews</td>
<td>0.86</td>
<td>0.47</td>
<td>0.24</td>
<td>0.2</td>
<td>0.75</td>
<td>1</td>
<td>0.31</td>
<td>0.36</td>
</tr>
<tr>
<td>FileSubmission</td>
<td>0.71</td>
<td>-0.13</td>
<td>0.96</td>
<td>0.35</td>
<td>0.45</td>
<td>0.31</td>
<td>1</td>
<td>0.18</td>
</tr>
<tr>
<td>GradeViews</td>
<td>0.39</td>
<td>0.12</td>
<td>0.21</td>
<td>0.48</td>
<td>0.58</td>
<td>0.36</td>
<td>0.18</td>
<td>1</td>
</tr>
</tbody>
</table>

**Fig. 19.** Plot of Correlation Matrix of Class 2 Log Data

### Positive Correlation between Visits and Course Views of Class 1

**Fig. 20.** Scatter Plot of Positive Correlation between Visits and Course Views of Class 1
Fig. 21. Scatter Plot of Positive Correlation between Assignments and File Submission of Class 1

Fig. 22. Scatter Plot of Positive Correlation between Visits and Course Views of Class 2
Fig. 23. Scatter Plot of Positive Correlation between Assignments and File Submissions of Class 2

Table 6. Clustering of Students of Class 1 Using K-Means Clustering Technique

<table>
<thead>
<tr>
<th>Class 1 Students</th>
<th>Cluster 2</th>
<th>Cluster 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Akila B</td>
<td>Alamelu Mangai R</td>
<td>Anjelin Jenifer G</td>
</tr>
<tr>
<td>Devadharshini M</td>
<td>Almas Banu A</td>
<td>Anusha J</td>
</tr>
<tr>
<td>Indu K</td>
<td>Bharathi J</td>
<td>Archana Devi T</td>
</tr>
<tr>
<td>Mohanavadhana T</td>
<td>Deepika K</td>
<td>Arthi R</td>
</tr>
<tr>
<td>Sowmiya D</td>
<td>Kavitha N</td>
<td>Aswathy N</td>
</tr>
<tr>
<td>Sri Seganithi V</td>
<td>Kavitha R</td>
<td>Deivanai K</td>
</tr>
<tr>
<td>Keerthana S</td>
<td>Divya SK Nair</td>
<td></td>
</tr>
<tr>
<td>Kousalya T</td>
<td>Girija Uma C</td>
<td></td>
</tr>
<tr>
<td>Malathi K</td>
<td>Karishma D</td>
<td></td>
</tr>
<tr>
<td>Monisha R</td>
<td>Lakshmi S</td>
<td></td>
</tr>
<tr>
<td>Narmatha M</td>
<td>Lavanya T</td>
<td></td>
</tr>
<tr>
<td>Nirmala M</td>
<td>Megala P</td>
<td></td>
</tr>
<tr>
<td>Ragavi K</td>
<td>Muthulakshmi E</td>
<td></td>
</tr>
<tr>
<td>Raja Chithra E</td>
<td>Nagaralakshmi K</td>
<td></td>
</tr>
<tr>
<td>Rajna R</td>
<td>Neeladevi V</td>
<td></td>
</tr>
<tr>
<td>Shabana J</td>
<td>Preethi G</td>
<td></td>
</tr>
<tr>
<td>Shanmugapiya A</td>
<td>Ranjisha B</td>
<td></td>
</tr>
<tr>
<td>Sreeja S</td>
<td>Revitha R</td>
<td></td>
</tr>
<tr>
<td>Swathi R</td>
<td>Sowmiya P</td>
<td></td>
</tr>
<tr>
<td>Vimala Mary R</td>
<td>Suganya R</td>
<td></td>
</tr>
<tr>
<td>Vimala Mary R</td>
<td>Tamil Amudhu R</td>
<td></td>
</tr>
<tr>
<td>Vimala Mary R</td>
<td>Thenmozhi R</td>
<td></td>
</tr>
</tbody>
</table>
Fig. 24. Scatter Plot of K-Means Clustering of Visits Vs Course Views of Class 1

Fig. 25. Scatter Plot of K-Means Clustering of Visits Vs Quizzes of Class 1
Fig. 26. Scatter Plot of K-Means Clustering of Visits Vs Assignments of Class

Fig. 27. Scatter Plot of K-Means Clustering of Visits Vs Course Views of Class 2
### Table 7. Frequency Range of the Attributes of Class 1 Log Data in the Clusters

<table>
<thead>
<tr>
<th>Class 1</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th>Cluster 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visits</td>
<td>106 to 131</td>
<td>63 to 93</td>
<td>35 to 62</td>
</tr>
<tr>
<td>Quizzes</td>
<td>3 to 5</td>
<td>1 to 4</td>
<td>1 to 3</td>
</tr>
<tr>
<td>Assignments</td>
<td>4 to 6</td>
<td>4 to 7</td>
<td>3 to 5</td>
</tr>
<tr>
<td>ForumRead</td>
<td>2 to 3</td>
<td>0 to 5</td>
<td>0 to 2</td>
</tr>
<tr>
<td>CourseViews</td>
<td>31 to 44</td>
<td>11 to 30</td>
<td>1 to 15</td>
</tr>
<tr>
<td>FileSubmission</td>
<td>6 to 9</td>
<td>5 to 10</td>
<td>5 to 8</td>
</tr>
</tbody>
</table>

### Table 8. Clustering of Students of Class 2 Using K-Means Clustering Technique

<table>
<thead>
<tr>
<th>Class 2</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abinaya C</td>
<td>Abinaya S</td>
<td></td>
</tr>
<tr>
<td>Aishwarya V</td>
<td>Ananthi A</td>
<td></td>
</tr>
<tr>
<td>Amirtha Varshini V</td>
<td>Anitha R</td>
<td></td>
</tr>
<tr>
<td>Anisha S</td>
<td>Deepika B</td>
<td></td>
</tr>
<tr>
<td>Kiruthika M</td>
<td>Ezhilmathy V</td>
<td></td>
</tr>
<tr>
<td>Nishanthi A</td>
<td>Gokulpriya B</td>
<td></td>
</tr>
<tr>
<td>Sankareswari S</td>
<td>Gowthami C</td>
<td></td>
</tr>
<tr>
<td>Sathya Roopa B</td>
<td>Hamsini Krishna S</td>
<td></td>
</tr>
</tbody>
</table>

**Fig. 28.** Scatter Plot of K-Means Clustering of Visits Vs Quizzes of Class 2
Fig. 29. Scatter Plot of K-Means Clustering of Visits Vs Assignments of Class 2

Table 9. Frequency Range of the Attributes of Class 2 Log Data in the Clusters

<table>
<thead>
<tr>
<th>Class 2</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visits</td>
<td>16 to 47</td>
<td>51 to 87, 125</td>
</tr>
<tr>
<td>Quizes</td>
<td>0 to 2</td>
<td>1 to 4</td>
</tr>
<tr>
<td>Assignments</td>
<td>1 to 4</td>
<td>2 to 6</td>
</tr>
<tr>
<td>ForumRead</td>
<td>0 to 1</td>
<td>1 to 4</td>
</tr>
<tr>
<td>CourseViews</td>
<td>1 to 8</td>
<td>8 to 28</td>
</tr>
<tr>
<td>FileSubmission</td>
<td>2 to 8</td>
<td>4 to 14</td>
</tr>
</tbody>
</table>

Table 10. Prediction of ctree Classification Technique for Training Dataset

<table>
<thead>
<tr>
<th>GRADE</th>
<th>FAIL</th>
<th>GOOD</th>
<th>EXCELLENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAIL</td>
<td>38</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>GOOD</td>
<td>0</td>
<td>36</td>
<td>5</td>
</tr>
<tr>
<td>EXCELLENT</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Response: Grade

Inputs: Quizes, Visits, Assignment, CourseViews, FileSubmission

Number of observations: 79

1) Quizes == {LOW}; criterion = 1, statistic = 156
   2) weights = 38

1) Quizes == {MEDIUM, HIGH}

3) CourseViews == {LOW}; criterion = 0.986, statistic = 40
   4) weights = 25

3) CourseViews == {MEDIUM, HIGH}
   5) weights = 16

Table 11. Prediction of ctree Classification Technique for Test Dataset

<table>
<thead>
<tr>
<th>GRADE</th>
<th>FAIL</th>
<th>GOOD</th>
<th>EXCELLENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAIL</td>
<td>9</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>GOOD</td>
<td>0</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>EXCELLENT</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
**Table 12.** Association Rules Produced on the Entire Log Dataset by Apriori Algorithm

<table>
<thead>
<tr>
<th>LHS</th>
<th>RHS</th>
<th>Support</th>
<th>Confidence</th>
<th>Lift</th>
</tr>
</thead>
<tbody>
<tr>
<td>{Visits=LOW, FileSubmission=LOW}</td>
<td>=&gt; {Assignment=LOW}</td>
<td>0.1</td>
<td>0.9</td>
<td>6.2</td>
</tr>
<tr>
<td>{CourseViews=LOW, FileSubmission=LOW}</td>
<td>=&gt; {Assignment=LOW}</td>
<td>0.1</td>
<td>0.9</td>
<td>5.8</td>
</tr>
<tr>
<td>{Assignment=LOW}</td>
<td>=&gt; {FileSubmission=LOW}</td>
<td>0.1</td>
<td>0.9</td>
<td>5.1</td>
</tr>
<tr>
<td>{Quizes=MEDIUM}</td>
<td>=&gt; {Grade=GOOD}</td>
<td>0.4</td>
<td>1</td>
<td>2.3</td>
</tr>
<tr>
<td>{CourseViews=MEDIUM, FileSubmission=MEDIUM}</td>
<td>=&gt; {Visits=MEDIUM}</td>
<td>0.1</td>
<td>1</td>
<td>2.1</td>
</tr>
<tr>
<td>{CourseViews=MEDIUM}</td>
<td>=&gt; {Visits=MEDIUM}</td>
<td>0.2</td>
<td>0.9</td>
<td>2</td>
</tr>
<tr>
<td>{Quizes=LOW}</td>
<td>=&gt; {Grade=FAIL}</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>{Assignment=LOW, CourseViews=LOW}</td>
<td>=&gt; {Visits=LOW}</td>
<td>0.1</td>
<td>0.9</td>
<td>1.9</td>
</tr>
<tr>
<td>{CourseViews=MEDIUM, CourseViews=LOW}</td>
<td>=&gt; {Visits=LOW}</td>
<td>0.1</td>
<td>0.9</td>
<td>1.9</td>
</tr>
<tr>
<td>{Visits=LOW}</td>
<td>=&gt; {CourseViews=LOW}</td>
<td>0.5</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{CourseViews=MEDIUM, FileSubmission=MEDIUM}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.1</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{Quizes=MEDIUM, FileSubmission=MEDIUM}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.3</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{FileSubmission=MEDIUM, Grade=GOOD}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.3</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{Visits=LOW, Assignment=MEDIUM}</td>
<td>=&gt; {FileSubmission=MEDIUM}</td>
<td>0.3</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{Quizes=MEDIUM, Assignment=MEDIUM}</td>
<td>=&gt; {FileSubmission=MEDIUM}</td>
<td>0.4</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{Assignment=MEDIUM, Grade=FAIL}</td>
<td>=&gt; {FileSubmission=MEDIUM}</td>
<td>0.4</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{Assignment=MEDIUM, CourseViews=LOW}</td>
<td>=&gt; {FileSubmission=MEDIUM}</td>
<td>0.5</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>{Assignment=LOW}</td>
<td>=&gt; {CourseViews=LOW}</td>
<td>0.1</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>{CourseViews=MEDIUM}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.2</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>{Quizes=LOW}</td>
<td>=&gt; {CourseViews=LOW}</td>
<td>0.4</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>{Grade=FAIL}</td>
<td>=&gt; {CourseViews=LOW}</td>
<td>0.4</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>{FileSubmission=MEDIUM}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.7</td>
<td>1</td>
<td>1.2</td>
</tr>
<tr>
<td>{Visits=MEDIUM, Quizes=MEDIUM}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.3</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>{Visits=MEDIUM, Grade=GOOD}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.3</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>{Visits=MEDIUM}</td>
<td>=&gt; {FileSubmission=MEDIUM}</td>
<td>0.1</td>
<td>0.8</td>
<td>1.1</td>
</tr>
<tr>
<td>{FileSubmission=LOW}</td>
<td>=&gt; {CourseViews=LOW}</td>
<td>0.1</td>
<td>0.8</td>
<td>1.1</td>
</tr>
<tr>
<td>{Quizes=MEDIUM}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.4</td>
<td>0.9</td>
<td>1.1</td>
</tr>
<tr>
<td>{Grade=GOOD}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.4</td>
<td>0.9</td>
<td>1.1</td>
</tr>
<tr>
<td>{Visits=MEDIUM}</td>
<td>=&gt; {Assignment=MEDIUM}</td>
<td>0.4</td>
<td>0.8</td>
<td>1.1</td>
</tr>
<tr>
<td>{Visits=MEDIUM, Quizes=MEDIUM}</td>
<td>=&gt; {FileSubmission=MEDIUM}</td>
<td>0.2</td>
<td>0.9</td>
<td>1.1</td>
</tr>
<tr>
<td>{Visits=MEDIUM, Grade=GOOD}</td>
<td>=&gt; {FileSubmission=MEDIUM}</td>
<td>0.2</td>
<td>0.9</td>
<td>1.1</td>
</tr>
</tbody>
</table>
Fig. 31. Grouped Matrix of the Rules Produced by the Apriori Algorithm on the Entire Log Dataset

Discussions

The discussions of the above results are as listed below.

- In Class 1 the student Mohanavadhana. T., has the highest visit frequency and in Class 2 the student Abinaya. C., has the highest visit frequency (Fig. 2 and Fig. 3).
- In Class 1 the student Indu. K., has the highest quizz frequency and in Class 2 the student Subashini. B., has the highest quizz frequency (Fig. 4 & Fig. 5).
- In Class 1 the student Preethi G., has the highest Assignment frequency (Fig. 6).
- In Class 2 the students Tejesvini Ramesh, Sankareswari. S. and Anisha. S., has the highest Assignment frequency (Fig. 7).
- In Class 1 the student Anusha J., alone has created forum and no other student was involved in this task (Fig. 8).
- In Class 2 the student Tejesvini Ramesh, alone has created forum and no other student was involved in this task (Fig. 9).
- In Class 1 the student Girija Uma C., has the highest frequency of Forums Read and out of the 48 student, 26 students have not used the forums (Fig. 10).
- In Class 2 the student Tejesvini Ramesh, has the highest frequency of Forums Read and out of the 46 student, 31 students have not used the forums (Fig. 11).
- In Class 1 the student Mohanavadhana. T., has the highest frequency of Courses Viewed and the students Swathi R., Raja Chithra E. and Alamelu Mangai R., have the lowest frequency of Courses Viewed (Fig. 12).
- In Class 2 the student Abinaya C., has the highest frequency of Courses Viewed and the students Yuvasree D., Saranya B., Sankareswari S., Priyanka G., Prasanna Lakshmi S., Periyanyakam P., Nikisha B., Meena M. and Kowsalya C., have the lowest frequency of Courses Viewed (Fig. 13).
- In Class 1 the File Submission Frequency ranges from 5 to 10. The student Suganya R., has the highest File Submission frequency and the students Thenmozhi R, Sowmya P., Malathi K. and Kowsalya T. have the least frequency of file submissions (Fig. 14).
- In Class 2 the File Submission Frequency ranges from 2 to 14. The student Abinaya C., has the highest File Submission frequency and the students Yuvasree D, Hamsini Krishna S., and Gowthami C. have the least frequency of file submissions (Fig. 15).
• In Class 1 the student Divya SK Nair, has the highest Grade Views (6) and 30 out of 48 students have not viewed their grades even once (Fig. 16).

• In Class 2 only 5 students have viewed their grades and the student Vaishnavi M, has the highest Grade Views (4) and 41 out of 46 students have not viewed their grades even once (Fig. 17).

• From the Correlation analysis of Class 1 log data, we can observe two prominent positive correlations 1) Visits Vs Course Views 2) Assignment Vs File Submission both with 0.94 correlation coefficient (Fig. 18).

• From the Correlation analysis of Class 2 log data, we can observe two prominent positive correlations 1) Assignment Vs File Submission 2) Visits Vs Course Views both with 0.96 and 0.85 correlation coefficients respectively (Fig. 19).

• From the Table 5., we can see that in Class 1, there are three clusters of students using K-Means Clustering. Out of the 48 students, 6 students belong to cluster 1, 20 students belong to cluster 2 and 22 students belong to cluster 3.

• The Table 6., shows the range of frequencies of visits, quizzes, assignments, forum reads, course views and file submissions for each of the three clusters of Class 1 students.

• From the Table 7., we can see that in Class 2, there are two clusters of students using K-Means Clustering. Out of the 46 students, 34 students belong to cluster 1 and 12 students belong to cluster 2.

• The Table 8., shows the range of frequencies of visits, quizzes, assignments, forum reads, course views and file submissions for each of the two clusters of Class 2 students.

• The entire log dataset containing both the class data is split into training dataset and test dataset with 80% and 20% ratio respectively. The Grades of the students are categorized as EXCELLENT, GOOD and FAIL.

• The decision tree classification technique is applied on the training data set and the results of this prediction are shown in Table 9 and Fig 50. The decision tree in Fig. 50 show that if a student takes LOW number of Quizzes, she will have FAIL Grade, if the student takes MEDIUM or HIGH number of Quizzes, and if her Course Views is LOW, she will get GOOD Grade and if the student takes MEDIUM or HIGH number of Quizzes, and if her Course Views is MEDIUM or HIGH, the student will get GOOD or EXCELLENT Grade.

• The prediction of the test dataset is shown in the Table 10 and it shows that the ctree classification technique correctly classifies 9 students with FAIL Grade and 5 students with GOOD Grade. But, the technique wrongly classifies the student with EXCELLENT Grade as with GOOD Grade.

• The Table 12., shows the 32 rules produced by the Apriori Algorithm taking the entire log dataset as input. These rules are sorted based on the descending values of Lift. There are two prominent rules that tells the instructor that if the Quizzes = MEDIUM, then we have Grade = GOOD and also if Quizzes = LOW, then we have Grade = FAIL. There are also rules that states the reason for Assignment = LOW (Rules [1], [2]) and Assignment = MEDIUM ([12], [13], [14], [20], [23], [24], [25], [28], [29], [30]). From these rules the instructor can learn the pattern and alert the student on time to score high and pass the course.

V. CONCLUSIONS

There are many Learning Analytics tools that are based on Moodle data and they only provide basic analytics and graphs about the student’ interaction in the system. However, it is difficult to find any analytics tool that enables varied statistical and data mining techniques put together to extract knowledge about the course performance. Hence, the focus of this work is to propose and prototype an analytics framework that uses several integrated data science techniques for discovering knowledge from Moodle data.

The main highlight of the proposed framework is that it uses the data analytics techniques available in the R Open Source Software. The proposed framework can be considered as a complement for all existing Learning Analytics tools, providing new, and varied functionalities related to the knowledge discovery process: preprocessing, data transformation, statistical techniques and important data mining techniques like clustering, classification and association rule mining.

In order to show case the usefulness of the proposed tool, we have designed a prototype of the proposed model and produced results based on sample Moodle data. The use of the proposed model is described with a case study using a sample real-life dataset of computer science students. The results of this case study show how the framework enables input data to be transformed into the instructor’s requirement. The model and results obtained were analyzed and discussed in order to show its usefulness for providing instructors with the feedback about how students learn within Moodle courses.

As a future scope of this work, we intend to add new data mining algorithms in order to provide instructors with more advanced insights. We shall explore the use of other clustering techniques other than K-Means, other classification techniques other than decision trees and improved version of Apriori algorithm for association rule mining. We also propose to work on more efficient visualization techniques available in the R Package and present the knowledge in an effective manner for better post processing.
REFERENCES


The coding used in R Studio to obtain the results of this work:

```r
logdata <- read.csv("logs_20170523-1240.csv")
logdata <- logdata[logdata$User.full.name != ".",]
logdata <- logdata[logdata$User.full.name != "Admin ADU",]
logdata <- logdata[logdata$User.full.name != "Guest user ",]
logdata <- logdata[logdata$Origin != "cli",]
logdata <- logdata[logdata$Component != "System",]
Timestamp <- as.POSIXlt(strptime(logdata$Time, "%d/%m/%y, %H"))
Username <- logdata$User.full.name
Component <- logdata$Component
Event <- logdata$Event.name
Ipadd <- logdata$IP.address

log_prep <- data.frame(Timestamp, Username, Component, Event, Ipadd)
log_prep <- log_prep[!duplicated(log_prep),]

library(sqldf)

query <- "SELECT Username, Event, count(*) 'Event Count' FROM log_prep GROUP BY Username, Event"
user_event <- sqldf(query)

query <- "SELECT Username, count(*) 'Visits' FROM log_prep GROUP BY Username"
user_visit <- sqldf(query)

query <- "SELECT Username, count(*) 'Quizes' FROM log_prep WHERE Event = 'Quiz attempt submitted' GROUP BY Username"
user_quiz <- sqldf(query)

query <- "SELECT Username, count(*) 'Assignment' FROM log_prep WHERE Event = 'A submission has been submitted.' GROUP BY Username"
user_assignment <- sqldf(query)

query <- "SELECT Username, count(*) 'ForumCreated' FROM log_prep WHERE Event = 'Discussion subscription created' GROUP BY Username"
user_forum_created <- sqldf(query)

query <- "SELECT Username, count(*) 'ForumRead' FROM log_prep WHERE Event = 'Discussion viewed' GROUP BY Username"
user_forum_read <- sqldf(query)

query <- "SELECT Username, count(*) 'CourseViews' FROM log_prep WHERE Event = 'Course module viewed' GROUP BY Username"
user_course_view <- sqldf(query)

query <- "SELECT Username, count(*) 'FileSubmission' FROM log_prep WHERE Event = 'A file has been uploaded.' OR Event = 'An online text has been uploaded.' GROUP BY Username"
user_file_submit <- sqldf(query)

query <- "SELECT Username, count(*) 'GradeViews' FROM log_prep WHERE Event = 'Grade user report viewed' OR Event = 'Grade overview report viewed' GROUP BY Username"
user_grade_view <- sqldf(query)

query <- "SELECT a.Username, Visits, Quizes FROM user_visit a LEFT JOIN user_quiz b ON a.Username = b.Username"
g1 <- sqldf(query)
g1[is.na(g1)] <- 0
```
query <- "SELECT a.Username, Visits, Quizes, Assignment FROM g1 a LEFT JOIN user_assignment b ON a.Username = b.Username"
g2 <- sqldf(query)
g2[is.na(g2)] <- 0
query <- "SELECT a.Username, Visits, Quizes, Assignment, ForumCreated FROM g2 a LEFT JOIN user_forum_created b ON a.Username = b.Username"
g3 <- sqldf(query)
g3[is.na(g3)] <- 0
query <- "SELECT a.Username, Visits, Quizes, Assignment, ForumCreated, ForumRead FROM g3 a LEFT JOIN user_forum_read b ON a.Username = b.Username"
g4 <- sqldf(query)
g4[is.na(g4)] <- 0
query <- "SELECT a.Username, Visits, Quizes, Assignment, ForumCreated, ForumRead, CourseViews FROM g4 a LEFT JOIN user_course_view b ON a.Username = b.Username"
g5 <- sqldf(query)
g5[is.na(g5)] <- 0
query <- "SELECT a.Username, Visits, Quizes, Assignment, ForumCreated, ForumRead, CourseViews, FileSubmission FROM g5 a LEFT JOIN user_file_submit b ON a.Username = b.Username"
g6 <- sqldf(query)
g6[is.na(g6)] <- 0
query <- "SELECT a Username, Visits, Quizes, Assignment, ForumCreated, ForumRead, CourseViews, FileSubmission, GradeViews FROM g6 a LEFT JOIN user_grade_view b ON a.Username = b.Username"
g7 <- sqldf(query)
g7[is.na(g7)] <- 0
dat <- g7
write.csv(dat, "F:/Elearning/dat.csv")
dat1 <- read.csv("dat1.csv")
dat2 <- read.csv("dat2.csv")
query <- "SELECT Username, Visits FROM dat1 WHERE Visits >= 35 AND Visits <= 65"
dat1lowvisit <- sqldf(query)
query <- "SELECT Username, Visits FROM dat1 WHERE Visits >= 66 AND Visits <= 99"
dat1medvisit <- sqldf(query)
query <- "SELECT Username, Visits FROM dat1 WHERE Visits >= 100 AND Visits <= 131"
dat1highvisit <- sqldf(query)
query <- "SELECT Username, Visits FROM dat2 WHERE Visits >= 16 AND Visits <= 52"
dat2lowvisit <- sqldf(query)
query <- "SELECT Username, Visits FROM dat2 WHERE Visits >= 53 AND Visits <= 89"
dat2medvisit <- sqldf(query)
query <- "SELECT Username, Visits FROM dat2 WHERE Visits >= 90 AND Visits <= 125"
dat2highvisit <- sqldf(query)
library(lattice)
barchart(dat1$Username ~ dat1$Visits, main = "Class 1 Visits", xlab = "Visit Frequency")
barchart(dat2$Username ~ dat2$Visits, main = "Class 2 Visits", xlab = "Visit Frequency")
barchart(dat1$Username ~ dat1$Quizes, main = "Class 1 Quizes", xlab = "Quiz Frequency")
barchart(dat2$Username ~ dat2$Quizes, main = "Class 2 Quizes", xlab = "Quiz Frequency")
barchart(dat1$Username ~ dat1$Assignment, main = "Class 1 Assignments", xlab = "Assignment Frequency")
barchart(dat2$Username ~ dat2$Assignment, main = "Class 2 Assignments", xlab = "Assignment Frequency")
barchart(dat1$Username ~ dat1$ForumCreated, main = "Class 1 Forums Created", xlab = "Forums Created Frequency")
barchart(dat2$Username ~ dat2$ForumCreated, main = "Class 2 Forums Created", xlab = "Forums Created Frequency")
barchart(dat1$Username ~ dat1$ForumRead, main = "Class 1 Forums Read", xlab = "Forums Read Frequency")
barchart(dat2$Username ~ dat2$ForumRead, main = "Class 2 Forums Read", xlab = "Forums Read Frequency")
barchart(dat1$Username ~ dat1$CourseViews, main = "Class 1 Courses Viewed", xlab = "Courses Viewed Frequency")
barchart(dat2$Username ~ dat2$CourseViews, main = "Class 2 Courses Viewed", xlab = "Courses Viewed Frequency")
barchart(dat1$Username ~ dat1$FileSubmission, main = "Class 1 File Submissions", xlab = "File Submission Frequency")
barchart(dat2$Username ~ dat2$FileSubmission, main = "Class 2 File Submissions", xlab = "File Submissions Frequency")
barchart(dat1$Username ~ dat1$GradeViews, main = "Class 1 Grade Views", xlab = "Grade View Frequency")
barchart(dat2$Username ~ dat2$GradeViews, main = "Class 2 Grade Views", xlab = "Grade View Frequency")
summary(dat1)
summary(dat2)
library("corrplot")
dat1num <- data.frame(dat1[,2:9])
dat2num <- data.frame(dat2[,2:9])
M<-cor(dat1num)
corrplot(M, method = "number", main = "Correlation Analysis of Class 1", cex = 0.5, mar = c(1,1,1,1))
N<-cor(dat2num)
corrplot(N, method = "number", main = "Correlation Analysis of Class 2", mar = c(1,1,1,1))
plot(dat1num$Visits, dat1num$CourseViews, main = "Positive Correlation between Visits and Course Views of Class 1", xlab = "Visits", ylab = "Course Views")
plot(dat1num$Assignment, dat1num$FileSubmission, main = "Positive Correlation between Assignments and File Submissions of Class 1", xlab = "Assignments", ylab = "File Submissions")
plot(dat1num$Visits, dat1num$CourseViews, main = "Positive Correlation between Visits and Course Views of Class 2", xlab = "Visits", ylab = "Course Views")
plot(dat2num$Assignment, dat2num$FileSubmission, main = "Positive Correlation between Assignments and File Submissions of Class 2", xlab = "Assignments", ylab = "File Submissions")
clust1 <- dat1
clust1$Username <- NULL
library(fpc)
pmk1 <- pamk(clust1)
pmk1$nc
km1 <- kmeans(clust1, 3)
table(dat1$Username, km1$cluster)
km1
plot(clust1[c("Visits", "CourseViews")], col = km1$cluster)
points(km1$centers[, c("Visits", "CourseViews")], col = 1:3, pch = 8, cex = 2)
plot(clust1[c("Visits", "Quizes")], col = km1$cluster)
points(km1$centers[, c("Visits", "Quizes")], col = 1:3, pch = 8, cex = 2)
plot(clust1[c("Visits", "Assignment")], col = km1$cluster)
points(km1$centers[, c("Visits", "Assignment")], col = 1:3, pch = 8, cex = 2)
clust2 <- dat2
clust2$Username <- NULL
pmk2 <- pamk(clust2)

km2 <- kmeans(clust2, 2)
table(dat2$Username, km2$cluster)

km2
plot(clust2[c("Visits", "CourseViews")], col = km2$cluster)
points(km2$centers[, c("Visits", "CourseViews")], col = 1:3, pch = 8, cex = 2)
plot(clust2[c("Visits", "Quizes")], col = km2$cluster)
points(km2$centers[, c("Visits", "Quizes")], col = 1:3, pch = 8, cex = 2)
plot(clust2[c("Visits", "Assignment")], col = km2$cluster)
points(km2$centers[, c("Visits", "Assignment")], col = 1:3, pch = 8, cex = 2)

query <- "SELECT Visits, Quizes, Assignment, CourseViews, FileSubmission, Grade FROM dat"
grade <- sqldf(query)

query1 <- "UPDATE grade SET Visits = 0 WHERE Visits < 50"
query2 <- "SELECT * FROM grade"
grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET Visits = 1 WHERE Visits >= 50 AND Visits < 100"
query2 <- "SELECT * FROM grade"
grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET Visits = 2 WHERE Visits >= 100"
query2 <- "SELECT * FROM grade"
grade <- sqldf(c(query1, query2))

grade$Visits <- as.factor(grade$Visits)
levels(grade$Visits) = c("LOW", "MEDIUM", "HIGH")
query1 <- "UPDATE grade SET Quizes = 0 WHERE Quizes = 0 OR Quizes = 1"
query2 <- "SELECT * FROM grade"
grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET Quizes = 2 WHERE Quizes = 4 OR Quizes = 5"
query2 <- "SELECT * FROM grade"
grade <- sqldf(c(query1, query2))

grade$Quizes <- as.factor(grade$Quizes)
levels(grade$Quizes) = c("LOW", "MEDIUM", "HIGH")
query1 <- "UPDATE grade SET Assignment = 0 WHERE Assignment = 1 OR Assignment = 2"
query2 <- "SELECT * FROM grade"
grade <- sqldf(c(query1, query2))
query1 <- "UPDATE grade SET Assignment = 1 WHERE Assignment = 3 OR Assignment = 4 OR Assignment = 5"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET Assignment = 2 WHERE Assignment = 6 OR Assignment = 7"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

grade$Assignment <- as.factor(grade$Assignment)
levels(grade$Assignment) = c("LOW", "MEDIUM", "HIGH")

query1 <- "UPDATE grade SET CourseViews = 0 WHERE CourseViews < 15"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET CourseViews = 1 WHERE CourseViews >= 15 AND CourseViews < 30"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

grade$CourseViews <- as.factor(grade$CourseViews)
levels(grade$CourseViews) = c("LOW", "MEDIUM", "HIGH")

query1 <- "UPDATE grade SET FileSubmission = 0 WHERE FileSubmission < 6"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET FileSubmission = 1 WHERE FileSubmission >= 6 AND FileSubmission < 10"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET FileSubmission = 2 WHERE FileSubmission >= 10"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

grade$FileSubmission <- as.factor(grade$FileSubmission)
levels(grade$FileSubmission) = c("LOW", "MEDIUM", "HIGH")

query1 <- "UPDATE grade SET Grade = 1 WHERE Quizes = 2 OR Quizes = 3"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET Grade = 2 WHERE Quizes = 4 OR Quizes = 5"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

query1 <- "UPDATE grade SET Quizes = 0 WHERE Quizes = 0 OR Quizes = 1"
query2 <- "SELECT * FROM grade"

grade <- sqldf(c(query1, query2))

grade$Grade <- as.factor(grade$Grade)
levels(grade$Grade) = c("FAIL", "GOOD", "EXCELLENT")
set.seed(1234)
i <- sample(2, nrow(grade), replace=TRUE, prob=c(0.8, 0.2))
train <- grade[i==1,]
test <- grade[i==2,]
form <- Grade ~ Quizes + Visits + Assignment + CourseViews + FileSubmission
library(party)
dt <- ctree(form, data=train)
table(predict(dt), train$Grade)
plot(dt)
dt
pred <- predict(dt, newdata = test)
table(pred, test$Grade)
library(arules)
rules.all <- apriori(grade)
quality(rules.all) <- round(quality(rules.all), digits=1)
rules.sorted <- sort(rules.all, by="lift")
inspect(rules.sorted)
subset.matrix <- is.subset(rules.sorted, rules.sorted)
subset.matrix[lower.tri(subset.matrix, diag=T)] <- FALSE
redundant <- colSums(subset.matrix, na.rm=T) >= 1
which(redundant)
rules.pruned <- rules.sorted[!redundant]
inpect(rules.pruned)
library(arulesViz)
plot(rules.pruned, method="grouped")