International Journal of Engineering Education Vol. 28, No. 3, pp. 744–754, 2012
Printed in Great Britain

0949-149X/91 $3.00+0.00
© 2012 TEMPUS Publications.

Parameter Free Visual Exploration Tool for Mining School Records*

TURGAY TUGAY BILGIN
Maltepe University, Dept. of Software Engineering, Maltepe, Istanbul, Turkey. E-mail: ttbilgin@maltepe.edu.tr

In this study a new framework especially designed for educational data mining has been proposed and named as Visual Cluster Exploration Framework (VCEF). It differs from existing alternatives such as parallel coordinates and icon based projections in terms of dealing with the curse of dimensionality, parameter free design and ease of use. The visualization subsystem of the VCEF employs a novel visualization technique which is called as SD-plots. The results in both synthetic and real life data sets demonstrate that the approach is highly effective and helps educators to discover clusters easily. The results could be used to organize well-balanced student groups to improve the active learning in engineering classes.

Keywords: data mining; visualization; clustering; educational data mining

1. Introduction

The incorporation of computers in most aspects financial, industrial, scientific and even daily activities has led to the collection of large amounts of data. Extracting the valuable information hidden in them is a difficult task. Data Mining tries to structure data and find inherent, possibly important, relations in the data. Data mining is the natural evolution of information technology and driven by a data-rich but information-poor situation [1].

Visual data mining is a novel approach to data mining [2]. It denotes the combination of traditional data mining techniques and information visualization methods. The utilization of both the automatic analysis methods and human perception and understanding promises more effective data mining techniques. Visual data mining techniques have proven to be of high value in exploratory data analysis and they also have a high potential for exploring large databases.

Data mining algorithms should have as few parameters as possible, ideally none. A parameter-free algorithm prevents analyst from imposing prejudices and presumptions on the problem at hand, and let the data itself speak [3]. The complex parameters of available analysis techniques make it difficult to comprehend and control the mining process; consequently, data mining tools are sometimes awkward to use and the results are difficult to value.

In this paper, a new visual data mining framework is proposed, which is especially designed for school records data sets. It is named as Visual Cluster Exploration Framework (VCEF). General conceptual background of this approach is the sorted dissimilarity plots (SD-Plots) and their implementations which are the main contributions of this study. VCEF differs from existing alternatives such as parallel coordinates and icon based projections in terms of dealing with the curse of dimensionality, simplicity and ease of use. Through VCEF, patterns could be visually better organized and have a higher chance to be revealed. To the best of knowledge, this is the first time that such an approach and tool has been proposed for educational data mining.

The rest of the paper is organized as follows. In Section 2, a short review is provided for the related work in visual cluster analysis and educational data mining. Section 3 includes the discussion about the problems that motivate this work. Section 4 presents the visual cluster analysis approach with theoretical background. Section 5 sketches the implementation process and illustrates how this data is used to help teachers and learners as well as its effectiveness on cluster exploration and cluster validation. Finally, the discussions, conclusions and further research are outlined.

2. Related work

Data mining requires the inclusion of the human in the data exploration process in order to be effective [4]. Visual data exploration is the process of presenting data in some visual form and allowing the human to interact with the data to create insightful representations [4]. Keim [4], and Keim and Kriegel [5] provide taxonomies of visualization-based data exploration approaches and note that these approaches can be classified by 1) the type of data, 2) the visualization technique, and 3) the interaction techniques. With the dramatic increase in the amount of data being captured by organizations, multidimensional visualization techniques have become an important area of data mining research.

* Accepted 16 January 2012.

Representing multidimensional data in a two- or three-dimensional visual graphic cannot be achieved through simple mapping. In the last decade, a large number of novel information visualization techniques have been developed, allowing visualizations of multidimensional data sets without inherent two- or three-dimensional semantics. Nice overviews of the approaches can be found in a number of recent books [6–8]. Visual data mining is a novel approach to deal with the growing flood of information. The aim is to combine traditional data mining algorithms with information visualization techniques to utilize the advantages of both approaches [9]. Ankerst [10] classifies current visual data mining approaches into three categories.

Widely used in business, data mining has scarce applications to education. The Educational Data Mining community website defines educational data mining (EDM) as follows [11]: 'Educational Data Mining is an emerging discipline, concerned with developing methods for exploring the unique types of data that come from educational settings, and using those methods to better understand students, and the settings which they learn in.' Educational Data Mining researchers study a variety of areas, including individual learning from educational software, computer supported collaborative learning, computer-adaptive testing and the factors that are associated with student failure or non-retention in courses. Across these domains, one key area of application has been in the improvement of student models [11]. Student models represent information about a student's characteristics or state, such as the student's current knowledge, motivation, meta-cognition, and attitudes. Modeling student individual differences in these areas enables software to respond to those individual differences, significantly improving student learning [12]. In recent years, researchers have used EDM methods to infer whether a student is gaming the system [13], experiencing poor self-efficacy [14] or even if a student is bored or frustrated [15].

Simple statistics, queries or visualization algorithms are useful to give to teachers/tutors an overall view of how a class is doing. More sophisticated information visualization techniques are used in [16] to externalize student data and generate pictorial representations for course instructors to explore. Using features extracted from log data and marks obtained in the final exam, some researchers use classification techniques to predict student performance fairly accurately [17]. These allow tutors to identify students at risk and provide advice ahead of the final exam [18]. When student mistakes are recorded, association rules algorithms can be used to find mistakes often associated together [19].

3. Motivation

A school has to use relevant data to improve student learning. Efficient use of data requires data tools, and such tools are necessary to provide student data to teachers when needed and without any delay for analysis.

There are many commercial excellent data mining platforms such as SAS Enterprise Miner [20] and SPSS Modeler [21] as well as open sources such as Weka [22], Knime [23], and RapidMiner [24]. However, they are not tailored to the needs for education. Firstly, these platforms are too complex for teachers and educators to use. Their features are well beyond the needs of teachers. In addition, the great majority of conventional clustering algorithms employed in data mining tools, generally require an appropriate model selection strategy due to their dependency on multiple parameters, which may be difficult to tune.

From this respect, this study proposes a visual cluster exploration framework to address the introduced problems. Basically, the framework integrates a scalable preprocessing step to organize high-dimensional information space with a new visualization technique for interactively discovering clusters and information structure.

4. Visual Cluster Exploration Framework (VCEF)

In order to solve the problems addressed in section 3, a new framework is suggested to integrate a scalable preprocessing step and a high dimensional visualization method. Basically, the preprocessing step Υ implements transformation of the high dimensional input space X into dissimilarity space $\boldsymbol{S}$ using Pearson correlation coefficient. $\boldsymbol{S}$ is transformed into row sorted dissimilarity matrix Ω by ψ step. In order to explore possible clusters in Ω matrix, the framework offers a new visualization technique called Sorted Dissimilarity Plots (SD-Plots). Λ step denotes plotting Ω matrix into a Zoomable User Interface (ZUI) for visually exploring clusters covered in section 4.2.

Figure 1 gives an overview of the proposed VCEF process. Let n be the number of samples in the data and d the number of attributes or dimensions for each sample x_j with $j \in \{1, \ldots, n\}$. The input data can be represented by n x d data matrix X with the j-th column vector representing the sample x_j. Dissimilarities are computed using Pearson correlation coefficient on Υ step that yields the n x n dissimilarity matrix $\boldsymbol{S}$. On ψ step, each row of $\boldsymbol{S}$ is incrementally sorted. The resulting Ω matrix is very useful for visualization. Since the Ω matrix is 2-dimensional, it

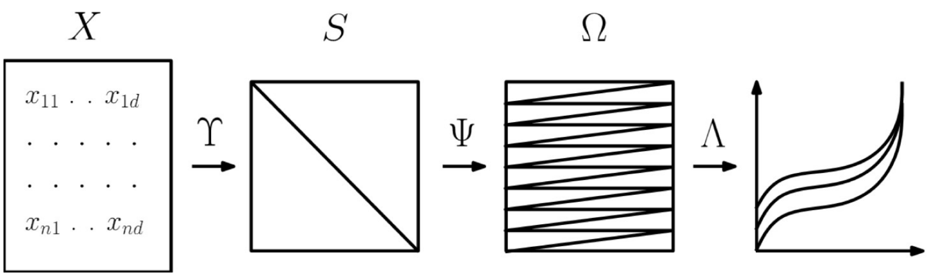


Fig. 1. Process flow diagram of the proposed Visual Cluster Exploration Framework.

can be readily line-by-line plotted on a 2D interactive zoomable plane. The visual cluster exploration process is described in section 4.2.

4.1 Transforming high dimensional space into distance matrix

Data mining algorithms heavily compute pairwise similarities. Generally, similarity computations are directly integrated into the clustering algorithms, which start straight from the original feature space. The key difference between the parallel coordinates and the proposed SD-plots is the focus on the similarity space instead of working directly in the feature domain F.

The similarity transformation $\Psi : \mathrm{F} \times \mathrm{F} \rightarrow \mathrm{S}$ translates a pair of internal and object-centered descriptions of features into an external and relationship-oriented space S. While there are n d-dimensional descriptions, there are also $(n-1)\,n/2$ pairwise relations [25]. Transforming d-dimensional data space into ***S*** enables one to work on a space which is independent of d. Therefore, the proposed approach does not suffer from the curse of dimensionality issue.

Distance measures dissimilarity [26], and dissimilarity measures the discrepancy between the two objects based on several features. Each similarity or dissimilarity metric has its own characteristics. In this study, dissimilarity (distance) was chosen for convenience rather than similarity. However, these are easily converted to each other.

There are many types of distance measures in the literature; however the perfect similarity measure doesn't exist yet. For geometrical distances, the Minkowski distance given by Equation (1) is a generalization of the well-known distance metrics.

$$d_{ij} = \sqrt[\lambda]{\sum_{k=1}^{n} \left| x_{ik} - x_{jk} \right|^{\lambda}} \tag{1}$$

For $\lambda = 1$, it is Manhattan distance, while it is Euclidean distance for $\lambda = 2$. Distance can also be defined by the cosine of the angle between two vectors. The cosine measure represents the similarity (s) rather than distance.

$$s_{ij} = \frac{\sum_{k=1}^{n} x_{ik} \cdot x_{jk}}{\left(\sum_{k=1}^{n} x_{ik}^{2} \sum_{r=1}^{n} x_{jr}^{2} \right)^{\frac{1}{2}}} \tag{2}$$

Therefore, higher values indicate that the two objects are similar. The value of cosine similarity falls within $[-1, 1]$ range.

Degree of dependence or correlation can be also used for defining similarity. Pearson correlation is the standardized cosine similarity by centering the coordinates to its mean value [26]. Pearson correlation is defined as

$$s_{ij} = \frac{\sum_{k=1}^{n} (x_{ik} - \bar{x}_i) \cdot (x_{jk} - \bar{x}_j)}{\left(\sum_{k=1}^{n} (x_{ik} - \bar{x}_i)^2 \sum_{r=1}^{n} (x_{jk} - \bar{x}_j)^2 \right)^{\frac{1}{2}}} \tag{3}$$

where

$$\bar{x}_i = \frac{1}{n} \sum_{k=1}^{n} x_{ik}$$

and

$$\bar{x}_j = \frac{1}{n} \sum_{k=1}^{n} x_{jk}. \tag{4}$$

The correlation values between -1 and $+1$ measure the degree of association within two variables. Pearson correlation also measures similarity rather than distance. The relationship between dissimilarity (distance) and similarity is given by

$$s_{ij} = 1 - \delta_{ij} \tag{5}$$

where normalized dissimilarity between object *i* and object *j* is denoted by δ_{ij}.

Pearson correlation distance is used in the study for creating distance matrix. Since the proposed tool focuses on student grades datasets, Euclidean distance performs poor on detecting covariance and trends in this case; therefore, Pearson Correlation coefficient is chosen to create SD-plots discussed in Section 4.2.

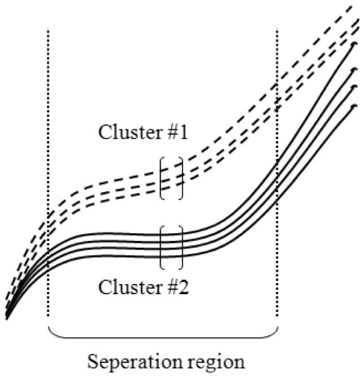


Fig. 2. SD-plots of a fictitious dataset containing two well separated clusters.

4.2 Sorted dissimilarity plots (SD-Plot)

One of the main contributions of this study is SD-plots. It is a simple yet effective visualization solution for the proposed framework. SD-plots map the n-dimensional space onto a two-dimensional plane. Let n be the number of samples in the data. For a given $n = 1$, a distance function is defined and each object is mapped to the distance from its 1st to n-th nearest neighbor. Sorting rows of distance matrix in the ascending order of their n-distance values and plotting each row onto a 2d plane give important information about the density distribution and clustering. These plots are called as the sorted dissimilarity plots. Fig. 2 demonstrates SD-plots of a fictitious dataset containing two well separated clusters. In SD-plots, the x-axis represents n-th neighbor of an object, and the y-axis represents distance. Visual cluster exploration is performed by observing the 'separation region' by the human perception. If there is no distinct separation region, then there is no cluster or there is a single cluster called singleton. In other words, visual cluster exploration means searching for separation regions on SD-plots.

The SD-plots implemented by using Open source C++ GUI library Qt [27]. Plot area is a Zoomable User Interface (ZUI) which is actually QtCanvas module of Qt Library. SD-plots are plotted in two steps:

- *Row sorting of dissimilarity matrix:* Each row of dissimilarity matrix (S) is sorted using generic qSort() algorithm [28] of Qt
- *Plot sorted rows*: the resulting Ω matrix is line-by-line plotted on the QtCanvas module.

When the separation region observed on the plots, bunches of lines are marked by the mouse pointer. The selected bunches are highlighted and the row number of highlighted lines are shown in the upper right window. It can be exported as a text file which is used as cluster labels of the dataset.

SD-plots have a number of advantages. Firstly, the plots are independent from ordering of the dataset. Secondly, they are easily recognized and interpreted by human eye. Thirdly, the visualization approach has no parameter and also the plots are also independent from the dimensionality. The next chapter demonstrates the efficiency of the approach.

5. VCEF implementation

VCEF is a tool for visualizing and interacting with the data based on a zoomable interface. One focus in the current implementation is to provide zooming to reduce cluttering. The important strategy is to visually explore the data space while zooming through it. A second focus is to design VCEF to make it relatively easy for teachers to use.

VCEF implementation fully supports the following features:

- The separation regions can be easily determined by mouse pointer on SD-plots (Fig. 3).
- If an SD-plot cannot fit to canvas area, it can be moved by using Ctrl+mouse keys.
- Selected part of an SD-plot can be zoomed by the mouse scroll.
- The index of the selected lines can be exported to a text file. It can be used as a cluster label index either for calculating clustering quality or exporting the labeled dataset to another data mining tool. For the convenience, the indexes of the selected lines are also displayed on a form placed beside the canvas.
- In order to reduce cluttering, selected bunches can be temporarily omitted from the canvas. There is also Undo and Redo buttons for user convenience.

VCEF is implemented in C++ with Qt GUI library. It is tested under various versions of the Unix operating system using the standard X graphics library system and Microsoft Windows operating system is also supported. It currently runs on a variety of BSD and Linux derivatives and Windows operating systems.

6. Experimental results

To demonstrate the efficiency of the framework, a number of case studies were performed using synthetic data sets as well as real world student grades data sets in this section.

6.1 Synthetic datasets

The algorithm was tested on synthetic datasets to more closely study various properties of the SD-plots approach. The Konstanz Information Miner's (KNIME) data generator module was used to

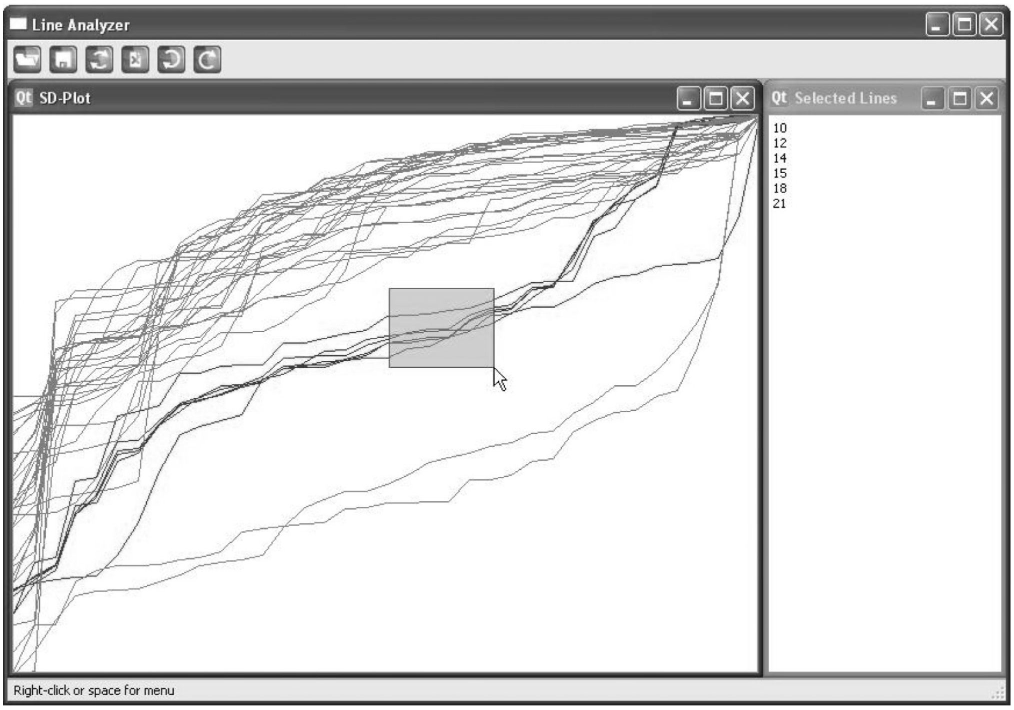


Fig. 3. Bunch selection using mouse pointer on VCEF tool.

Table 1. Parameter settings of synthetic datasets

Name	Syn_1	Syn_2	Syn_3
Type	Well-seperated	Fuzzy	High Dimensional
Samples (n)	75	75	300
Dimensions (d)	2	2	100
Standart Deviation (s)	0.03	0.10	0.20
Noise fraction (NF)	0.10	0.10	0.10
# of clusters	2	2	2

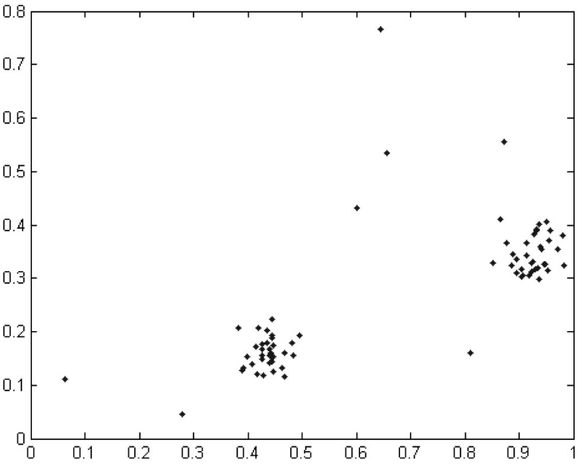


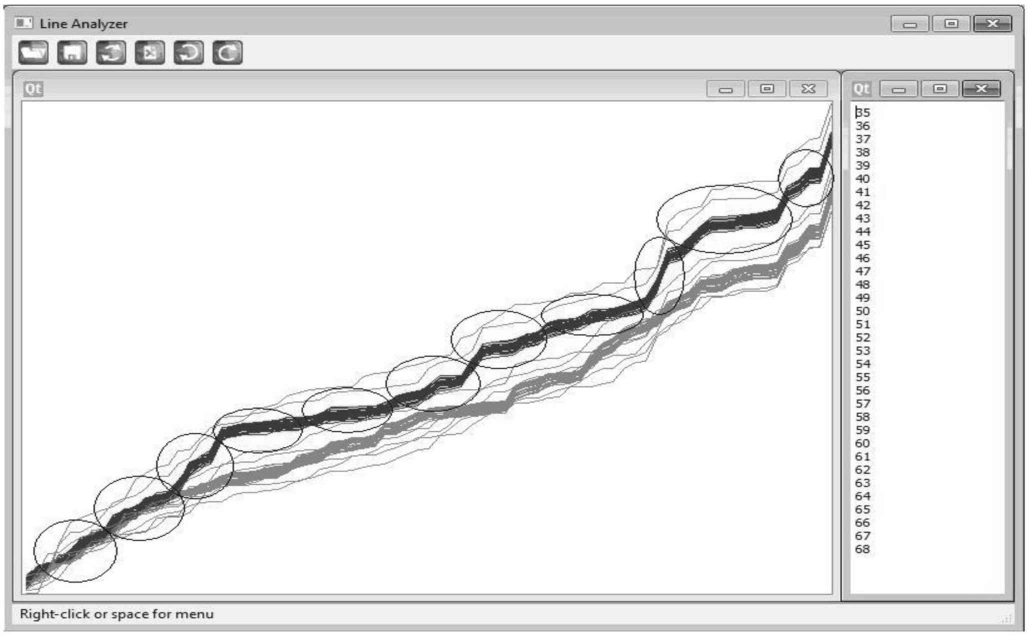
Fig. 4. 2-D synthetic dataset Syn_1 with two true clusters.

generate synthetic data sets with the parameters given in Table 1. KNIME [23] is a Java-based data mining platform with a graphical user interface based on Eclipse Project. Since they are generated randomly, no real correlation actually exists within samples. As a result, the correlation coefficient performs poor on synthetic datasets. Therefore, Euclidean metric was preferred for synthetic datasets.

Three experiments were performed to show the accuracy of the suggested approach. For the first experiment, a dataset was prepared with well-separated clusters and apparent outliers. Fig. 4 shows the scatter plot of Syn_1 and Fig. 5 shows the corresponding SD-plot.

Two clusters are readily recognized on the SD-plot. The first bunch (marked by the circles) represents the first cluster, and the index of cluster members are shown in the next window. The light gray bunch represents the second cluster. When it is highlighted by the mouse pointer, its member indexes will appear in the next window. The list of indexes can be exported as a text file for evaluating the accuracy and quality of clustering.

For the second experiment, the visual data mining tool was run on the second dataset Syn_2. It is two dimensional dataset with fuzzy clusters as shown in Fig. 6. The SD-plot of Syn_2 dataset is not as clear as the first dataset. Thanks to its zoomable canvas, the separation region was easily determined and



35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68

Fig. 5. A visually explored cluster on the SD-plot and index of the selected lines listed in the next window.

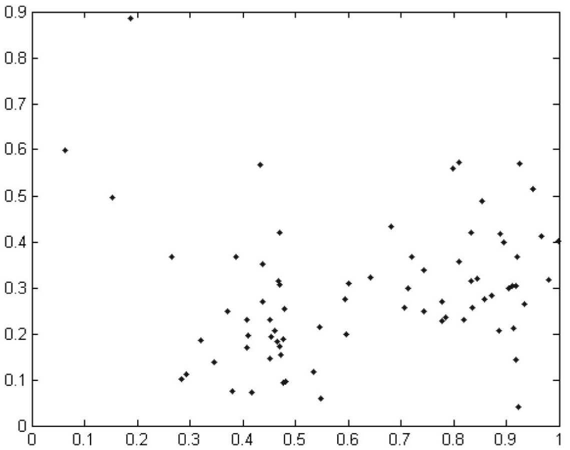


Fig. 6. Scatter plot of synthetic dataset Syn_2.

bunches were distinguished with the help of zooming as shown in Fig. 7. Fig. 8 shows the marked cluster on the SD-plot after zooming out.

Figure 9 shows a run of the VCEF tool on a high dimensional synthetic dataset with 100 dimensions, 1000 points and two true clusters using **s** = 0.2. Due to the high dimensionality, there is no 2-D scatter plot of Syn_2 dataset. The importance of the visualization tool is clearly seen in this case since there is no effective way of visualizing a 100 dimensional dataset.

VCEF tool is better at finding that the two clusters exist in 100 dimensional data space. One cluster marked by mouse pointer is seen on the Fig. 9. Above the selected cluster, the other bunch can be shown in grey. In VCEF tool, all clusters are not detected at once, and instead, user determines the clusters one by one. This behaviour reduces the cluttering.

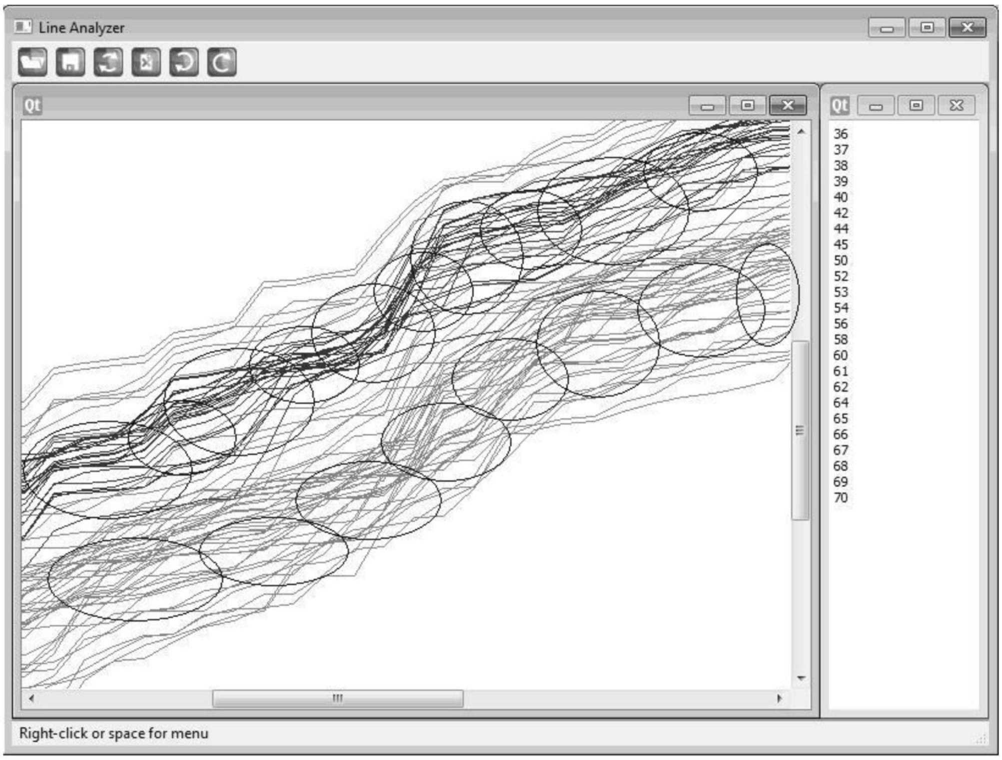
Figure 10 shows the zoomed SD-plot of separation region. It is very easy to figure out the separation region and explore two clusters on the figure. On various types of synthetic datasets, VCEF tool successfully visualizes them to help visual data exploration.

6.2 Real life datasets

In this part, two real life data sets were used. The first one contains '*Object Oriented Programming*' course grades of sophomore students from Maltepe University Software Engineering department. The data attributes are *student_id, midterm_1, midterm_2, term_project, pop_quiz, lab_assn,* and *final_exam* scores of year 2009. The 100-point grading scale is used in Maltepe University.

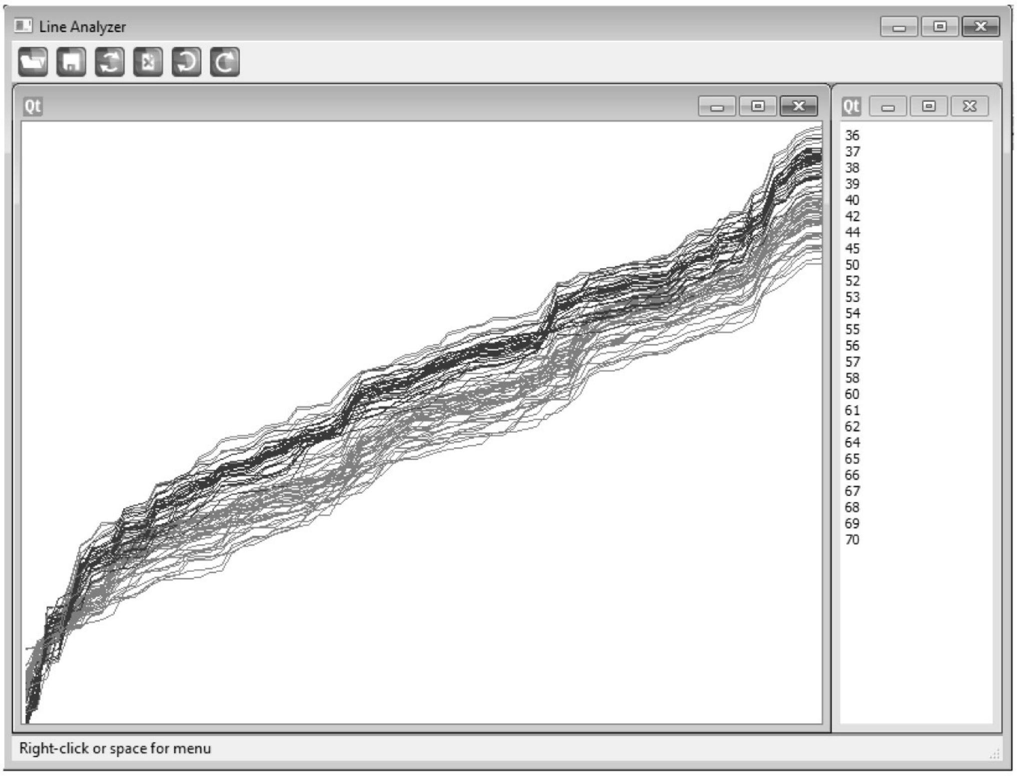
Figure 11 shows a run of the VCEF tool on the OOP course grades. The selection rectangle on the SD-plot points out a straightforward outliers cluster. The coherence within the cluster members is shown on Fig. 12. These two students got similar grades in the majority of the exams. The tool enables the teachers easily discover such students and helps to reveal underlying problems.

The second real life data set contains all course grades of senior class Computer Engineering students in Maltepe University. There are totally 79 compulsory and elective course grades for 50 students. The data set has zero values since some of the elective courses did not taken by the students. The corresponding SD-plot is shown on Fig. 13. A



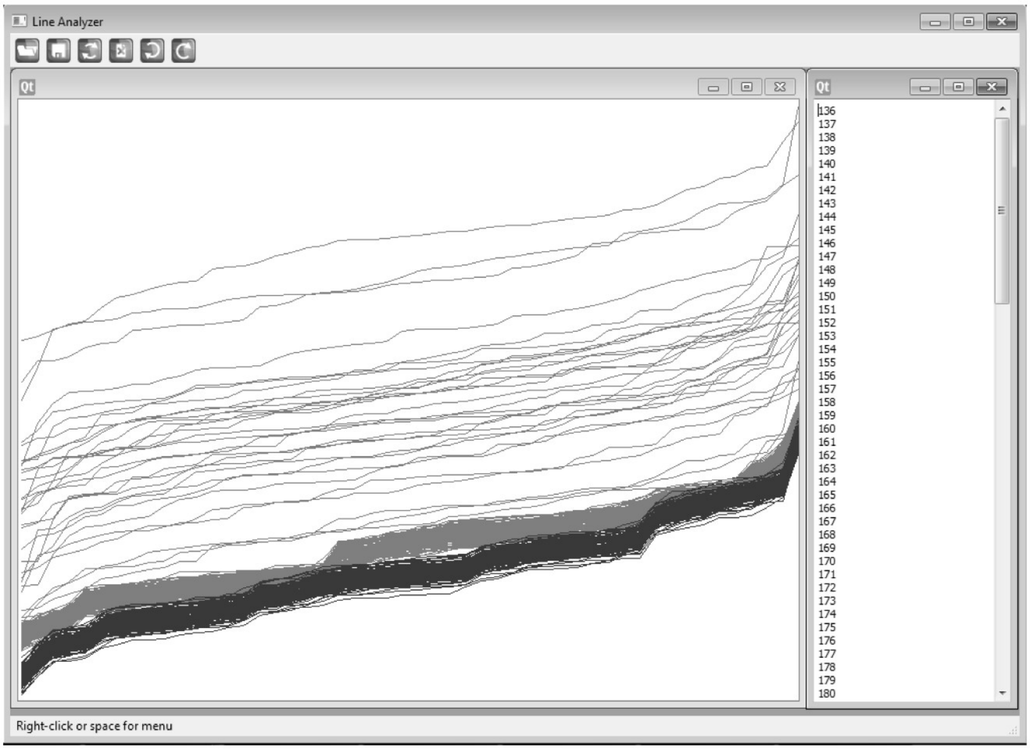
```
36
37
38
39
40
42
44
45
50
52
53
54
56
58
60
61
62
64
65
66
67
68
69
70
```

Fig. 7. Zooming in for determining separation region on noisy datasets (due to the grayscale image, two clusters are marked by circle arrays).



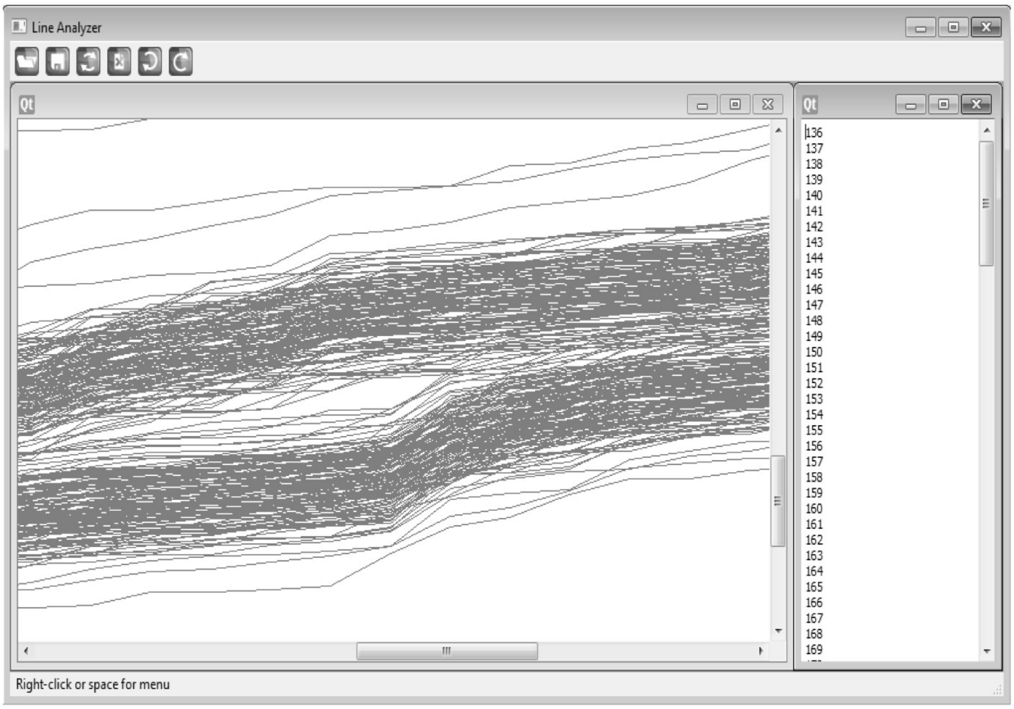
```
36
37
38
39
40
42
44
45
50
52
53
54
55
56
57
58
60
61
62
64
65
66
67
68
69
70
```

Fig. 8. A cluster marked on the noisy Syn_2 dataset (shown in dark color) and index of the selected lines listed in the next window.



```
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
```

Fig. 9. A cluster marked on the high dimensional Syn_3 dataset (shown in dark color) and index of the selected lines listed in the next window.



```
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
```

Fig. 10. Zooming in for determining separation region on high dimensional datasets.

cluster was discovered and marked in red on the plot.

The corresponding coherence plot is shown in Fig. 14. The x axis represents course ids and the y axis represents scores on the plot. There are 14 students in this cluster. They chose almost an identical set of electives and got similar scores on all courses of 4-year curriculum. There is a strong

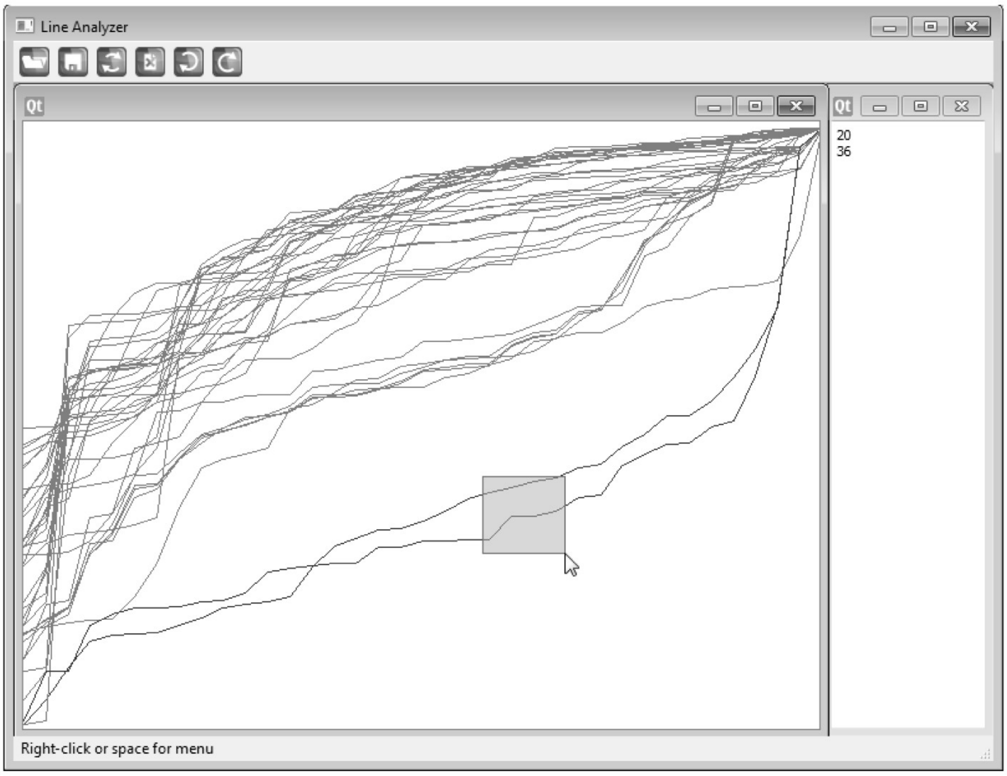


Fig. 11. A straightforward cluster on the SD-plot.

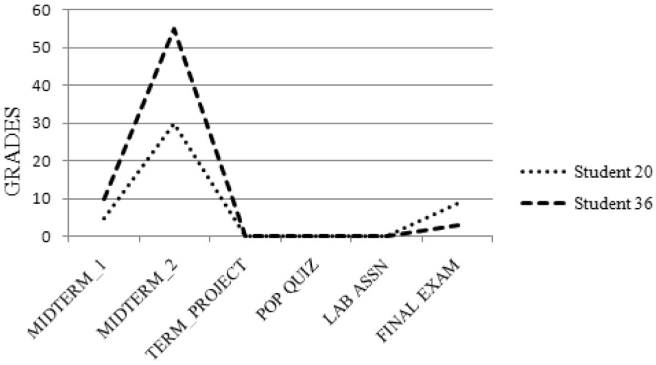


Fig. 12. The coherence plot for the selected bunch in Fig. 11.

correlation between the courses taken and scores obtained within the cluster.

This experiment is indicative of the power of the suggested approach. The tool helped to easily discover the clusters in the high dimensional real world data set. Further experiments were performed to determine effectiveness and accuracy of the approach. However, the results cannot be discussed here due to the page limitation.

7. Discussions

Student-centered teaching methods shift the focus of activity from the teacher to the learners. These methods include active learning, cooperative learning and inductive learning [29]. R.M. Felter explains active learning and demonstrates its use in engineering classes [29]. VCEF tools help the educators to organize well-balanced student groups. Therefore, it will improve the active learning and cooperative learning in engineering education.

There are two limitations of VCEF tool. First, when the number of objects grows up to thousands, the SD-plots get cluttered. In future work, visual optimizations such as colorizing and opacity to enhance the visualization will be studied. Second, VCEF tool cannot deal with the datasets containing categorical or rank data types, since some teachers

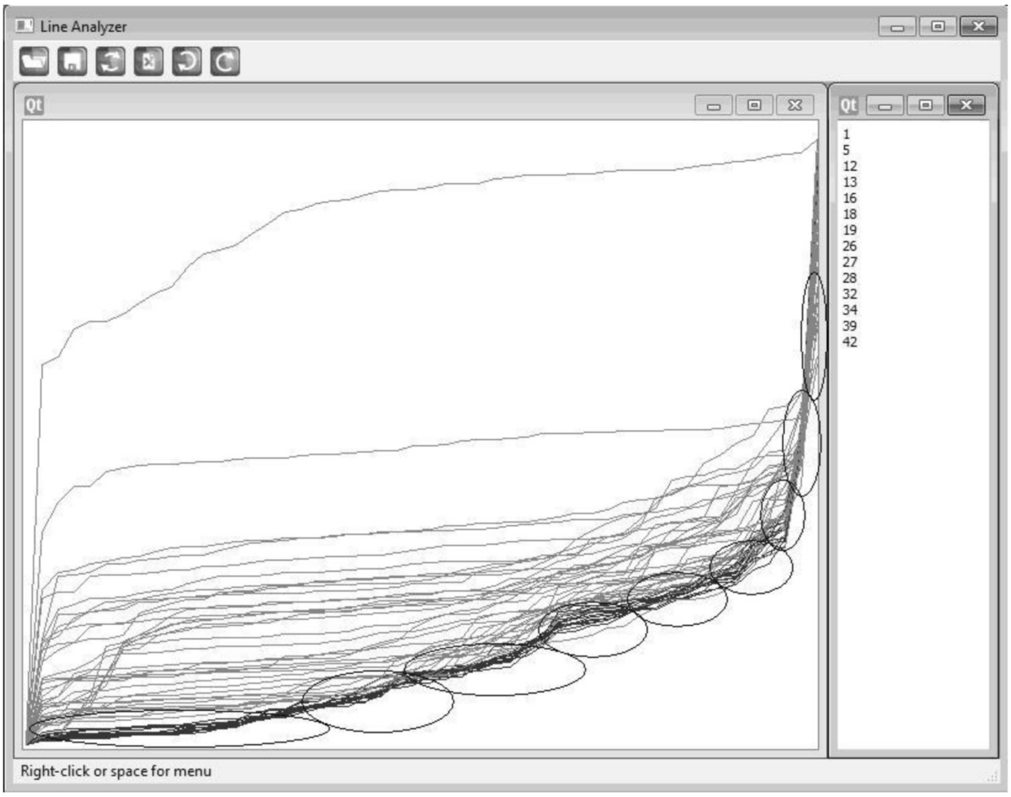


Fig. 13. A cluster discovered using SD-plot is marked in red (due to the grayscale image, two clusters are marked by circle arrays).

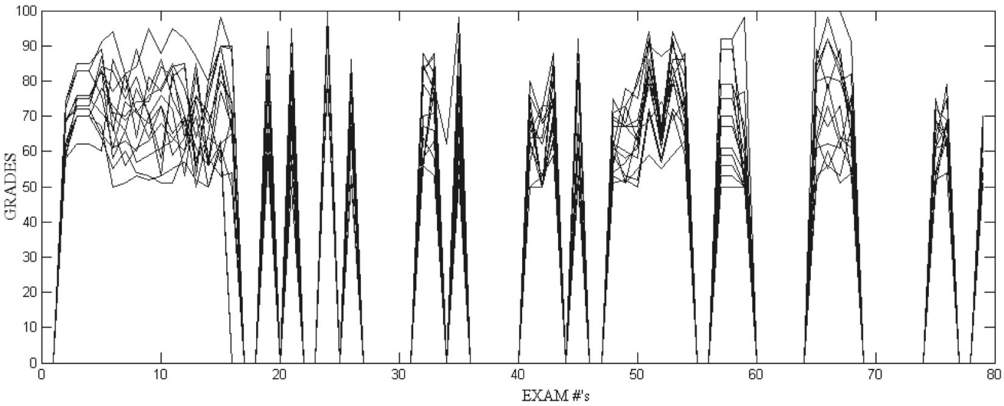


Fig. 14. The coherence plot for the selected bunch in Fig. 13.

classify the student grades as 'good', fair' and 'bad'. A better transformation function is currently being studied on to transform these types into the numerical equivalents.

It is aimed to extend the methodology employed in the study by proposing an approach for semi-automatically discovering the separation regions on the plots. Developing web based version of the VCEF tool is in progress. It will help web developers to integrate VCEF into a Learning Management System (LMS).

8. Conclusion

In this study, a new framework is defined, which is especially designed for educational data mining and intended to help teachers in the knowledge discovering process. A well-described data mining process and a novel visualization technique are main contributions of this work. The suggested solution differs from the alternatives in terms of dealing with the curse of dimensionality, simplicity, parameter-free design and ease of use.

Experimental results in both synthetic and real life data sets demonstrates that the proposed framework and visualization approach is highly effective and helps teachers to discover natural groupings among the students. The results could be used to assign students to groups to form well-balanced groups. The right group assignments in a cooperative learning environment can improve student learning.

Acknowledgements—I especially appreciate the support of Onur Tugcu as a guru programmer for enjoyable discussions about Qt library and zoomable interfaces.

References

1. J. Han and M. Kamber, *Data Mining: Concepts and Techniques, Second Edition*, Morgan Kaufmann, USA, 2006, pp. 3–11.
2. J. Simeon, M. H. Böhlen and A. Mazeika, Visual Data Mining: An Introduction and Overview, *Lecture Notes in Computer Science*, (4404), 2008, pp. 1–12.
3. E. Keogh, S. Lonardi and C. A. Ratanamahatana, Towards parameter-free data mining, *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining—KDD '04*, USA, 2004.
4. D. A. Keim, Information visualization and visual data mining, *IEEE Transactions on Visualization and Computer Graphics*, **8**(1), 2002, pp. 1–8.
5. D. A. Keim and H. P. Kriegel, Visualization techniques for mining large databases: A comparison, *IEEE Transactions on Knowledge and Data Engineering*, **8**(6), 1996, pp. 923–938.
6. C. Ware, *Information Visualization, Second Edition: Perception for Design*, Morgan Kaufmann, USA, 2004.
7. R. Spence, *Information Visualization: Design for Interaction, 2nd Edition*, Prentice Hall, USA, 2007.
8. R. Mazza, *Introduction to Information Visualization*, Springer Publishers, UK, 2009.
9. D. Keim, W. Müller and H. Schumann, *Visual Data Mining*, Eurographic STAR proceedings, Saarbrücken, Germany, 2002.
10. M. Ankerst, Visual Data Mining with Pixel-oriented Visualization Techniques, In: *Proceedings of ACM SIGKDD Workshop on Visual Data Mining '01*, USA, 2001.
11. R. Baker and K. Yacef, The state of educational data mining in 2009: A review and future visions, *Journal of Educational Data Mining*, **1**(1), 2009, pp. 3–17.
12. A. Corbett, Cognitive computer tutors: Solving the two-sigma problem, *User Modeling 2001*, Springer, 2001, pp. 137–147.
13. R. S. Baker, A.T. Corbett and K. R. Koedinger, Detecting student misuse of intelligent tutoring systems, In: *Proceedings of the 7th International Conference on Intelligent tutoring systems*, Brazil, 2004, pp. 531–540.
14. S. Mcquiggan, M. Mott and J. Lester, Modeling Self-Efficacy in Intelligent Tutoring Systems: An Inductive Approach, *User Modeling and User-Adapted Interaction*, **18**, 2008, pp. 81–123.
15. S. K. D'mello, S. D. Craig, A. W. Witherspoon, B. T. Mcdaniel and A. C. Graesser, Automatic Detection of Learner's Affect from Conversational Cues, *User Modeling and User-Adapted Interaction*, **18**, 2008, pp. 45–80.
16. R. Mazza and V. Dimitrova, CourseVis: Externalising Student Information to Facilitate Instructors in Distance Learning, In: *Proceedings of 11th International Conference on Artificial Intelligence in Education (AIED03)*, F. Verdejo and U. Hoppe (Eds), Sydney, 2003.
17. B. Minaei-Bidgoli, D. A. Kashy, G. Kortemeyer and W. F. Punch, Predicting student performance: An application of data mining methods with the educational web-based system LON-CAPA, *In Proceedings of ASEE/IEEE Frontiers in Education Conference*, USA, 2003.
18. A. Merceron and K. Yacef, Educational data mining: a case study, *Proceeding of the 2005 conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, Netherland, 2005, pp. 467–474.
19. A. Merceron and K. Yacef, A Web-based Tutoring Tool with Mining Facilities to Improve Learning and Teaching, *In Proceedings of 11th International Conference on Artificial Intelligence in Education*, F. Verdejo and U. Hoppe (Eds), Sydney, 2003, pp. 201–208.
20. Data Miner, http://www.sas.com/technologies/analytics/datamining/miner/, Accessed 14 April 2011.
21. SPSS Software modeler, http://www.spss.com/software/modeler/, Accessed 14 April 2011.
22. WEKA Tool, http://www.cs.waikato.ac.nz/ml/weka/, Accessed 14 April 2011.
23. KNIME Tool, http://www.knime.org/, Accessed 14 April 2011.
24. Rapid Miner, http://www.rapid-i.com/, Accessed 14 April 2011.
25. A. Strehl, *Relationship-based clustering and cluster ensembles for high-dimensional data mining*, PhD thesis, The University of Texas at Austin, 2002.
26. Similarity Metrics, http://people.revoledu.com/kardi/tutorial/Similarity/WhatIsSimilarity.html, Accessed 14 April 2011.
27. Qt Framework, http://qt.nokia.com/, Accessed 14 April, 2011.
28. Qt Generic algorithms, http://doc.qt.nokia.com/latest/qtalgorithms.html, Accessed 14 April, 2011.
29. R. M. Felder and R. Brent, Active Learning: An Introduction. *ASQ Higher Education Brief*, **2**(4), 2009.

T. Tugay Bilgin received the BSc, PhD degrees in Computer and Control Education from Istanbul Marmara University. His doctoral thesis was on the mining of high dimensional datasets. His research interests are high dimensional data mining, web mining, service oriented architecture and web services.