Clustering Users Based on the Capacity to Solve Questions in an Educational Platform

Mariana G. M. Macedo, and Carmelo J. A. Bastos-Filho

[1] Department of Computer Engineering – Polytechnic School of Pernambuco
Benfica Street, 455 - Madalena – University of Pernambuco (UPE)
CEP: 50720-001 – Recife – PE – Brazil

{mgmm,carmelofilho}@ecomp.poli.br

Abstract. *The intense daily use of educational platforms results in high volumes of data. Because of this gigantic pile of data, the users' needs frequently remain unnoticed and thus are not enhanced. This study characterised students' profiles based on an educational database. To achieve these goals, we assessed the application of the K-means and C-means algorithms for clustering. The number of profiles was chosen according to Davies-Bouldin metrics and Gap Statistic. After the experiments, K-means revealed to be inefficient for the database. On the other hand, C-means reached a satisfactory result, sustained by Spearman's rank correlation coefficient.*

Keywords*: Knowledge Discovery in Databases, Data Mining, Clustering, cluster, dissimilarities, Davies-Bouldin Metrics, Gap Statistic, Spearman's rank correlation coefficient.*

1. Introduction

Educational platforms are well accepted by the population due to its flexibility. Troubles related to officer hours, locality, heavy traffic, high costs and lack of certification could be resolved with an educational online platform. Platforms have been created for all kind of goals. Some platforms help the students to gather knowledge without a particular ambition. Some platforms propose to charge a modest amount of money to provide a specific material prepared for a particular exam. Moreover, some platform were created as an online environment to learn and to attend to a particular course, like undergraduate or graduate courses.

Independent of the primary function of the platform, the intense daily use of them lead to a high amount of data. The user's needs remain unnoticed by the platform because of the complexity to understand and identify the particularities in this gigantic pile of data. Nevertheless, the success of a platform depends on content teaching's efficiency and approaches to improve this can be helpful.

This study was encouraged to search for types of student in a particular educational platform, assigned in confidentiality by a private company. Considering user's profile knowledge, it is possible to identify the correlation between them and to improve the manner different contents are shown.

The educational platform used by this paper has several functionalities, but we selected the capability to solve questions. This feature can provide the efficiency by users in the presented subjects. Thus, it is possible to understand which type of students study more and have more interaction with the contents.

We used Knowledge Discovery in Databases (KDD) to analyse and identify types of students. In this process, the goal is to investigate the relevance and the meaning of the database. It is divided into five steps; the process studies how the data is stored, obtained, processed and used. In the first phase, named Data Selection, we choose which kind of data and characteristics have to be used. In the next step, the Data Pre-Processing, we analyse and modify discrepant, chronological and empty values. In the third phase, Data Transformation, we change the scale and type of values with functions or distributions. Data Mining, the fourth step, aims to select results with some predetermined goals. In this fourth step, the data is usually divided into groups: classify in sets, predict behaviour, detection of deviations and change. In the final step, the Data Interpretation, the results of all the steps are analysed and explained by documents, graphics, images, videos, presentations or any archive.

In this paper, we used the Data Mining's task of Clustering. Clustering is the process to divide a set into groups that have a similar characteristic, and each group has big differences each other. Cluster is the modern word used to identify a set grouped by Clustering task. In the final step, we used two metrics to choose the number of groups that minimises the dissimilarity intra-cluster and maximises the dissimilarity inter-cluster. The first one is the Davies-Bouldin Metrics. This technique gets the worst combination between each two groups and chooses the best number of groups that minimises this metrics. The second one is Gap Statistic that is used to find the best number of clusters that is far from some results samples chosen randomly.

We present a real users' profile analysis based on their efficiency in question resolution and amount of answered questions, both calculated for each subject. Furthermore, we explain the results of the algorithms (K-means and C-means) in a real database and the metrics to find the best number of clusters that has the biggest similarity inside groups and the smallest similarity between any two users of different groups. It is important to highlight that K-means and C-means were chosen since other more complex techniques, such as Self-Organizing Map, present a higher executing time, which is is not interesting for this application. SOM could be useful in case C-means is not efficient, which is not the case of this paper. Moreover, we have performed preliminaries tests with Self-Organizing Map and it lasted 20 hours more than the other techniques with comparable results.

This paper is divided into seven sections. In the Section 2, similar works of this paper are presented as a recommendation to further knowledge. Section 3 resumes and describes the paper contributions. Section 4 brings a summarized of K-means and C-means theory and how it was implemented. Section 5 explains Davies-Bouldin metrics and Gap Statistic and how they were used to select clusters' number that minimizes the difference between data inside the group and maximizes the difference between users in different groups. Section 6 presents the experiments realized to find the results displayed. Section 7 condenses the main achieved contributions of this paper.

2. Related works

Several works have been developed for clustering educational data mining. In 2010, many systems were developed to give hints in question resolutions [Dominguez et al. 2010]. Using distributional similarities and Dirichlet distribution, groups of students can be grouped by K-means [Gong et al. 2010]. Efficient starting centroids can be developed based on the Q-matrix to improve K-means algorithm convergence [Nugent et al. 2010]. In 2011, the addition of collaborative instructional

discussion activities in the platform is used for analysis [Huei-Tse 2011]. Each student can learn with several behaviors, [Kardan and Conati 2011] analyzed how it can be characterized. In 2012, K-means is used to distinct cognitive' student knowledge [Sparks et al. 2012]. The behavior of undergraduate students in the educational online platform is performed by analyzing the connection between answered exercise question and final grades [Abdous et al. 2012]. In [LEE 2012], students' abilities are analyzed based on their activities. In 2013, Learning patterns can be detected as beneficial [Malmberg et al. 2013]. In 2015, based on data density peaks, K-means initializes centroids by a new algorithm [Lan et al. 2015]. In 2016, special treatment of dispersion and overlap problems in data clustering are proposed in [Lin 2016].

3. Paper Contribution

The main contribution of this paper is to compare K-means and C-means algorithms in a real database to find users' profile in an online educational platform. By using the best algorithm between those algorithms in the confidential database utilized in this paper, we expect to detect several groups with a particular characteristic. Similar characteristics lead to significant aspects of the platform's students. Moreover, it is possible to identify which component does not determine the general student improvement. Another major contribution is the use of Davies-Bouldin Metrics and Gap Statistic to determine the number of clusters to separate different behavior of the students. These two metrics demonstrate good results of concerning the number of clusters even in real databases.

4. Algorithms

4.1. K-means

K-means algorithm remains popular because its simplicity and speed. It was developed and improved by Forgy [Forgy 1965] and Loyd [Lloyd 1982], MacQueen [MacQueen 1967] and Hartigan and Wong [Hartigan and Wong 1979]. Numerical, partitional, iterative and unsupervised, K-means is used to divide data in groups. The disadvantage of this algorithm is a suitability in finding global clusters and to split data in the boundary of clusters.

K-means minimizes the Equation (1) which k is groups's number, n_k is the number of patterns in the *cluster* k, c_k is cluster's centroid of k, i is position in the pattern, x_i is the pattern in position i and K is maximum number of pattern.

$$\epsilon = \sum_{k=1}^{K} \sum_{i=1}^{n_k} |x_i^k - c_k|^2. \quad (1)$$

All process of K-means can be summarized in 4 steps, two inputs and one output. Patterns' vector and the clusters' number are the inputs. A vector of patterns' grouped by the clusters' number is the output. The first step is the initialization of centroids. Until the stop criterion is not satisfied, we iterate the second, third and fourth steps. The second step calculates distances between all the patterns and centroids and reclassifies each pattern to the closest centroid. Third step updates each centroid by the Equation (2).

$$c_k = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i^k. \quad (2)$$

In this article, we used random initialization because it simplicity described by Peña, Lozano and Larrañaga [Peña et al. 1999]. Kaufman's technique (KA) [Kaufman and Rousseeuw 1990] has better results than random initialization but increases the complexity. Euclidean distance was used to calculate the distance between the values since all data has same the interval of values. The stop criterion used was the number maximum of iterations and the comparative of centers in iteration sequence.

4.2. C-means

Fuzzy C-means (FCM) has similar characteristics when compared to K-means. However, FCM uses Fuzzy Logic [Ghosh and Dubey 2013]. K-means does not group well data in boundaries. Thus, FCM can group those data using element's pertinence to all groups. Fuzzy C-means was idealized by James Bezdek [Bezdek 1981] and improved by Robert Ehrlich and William Full [Full et al. 1984]. It is utilized in several problems as Image Processing [Yong et al. 2004], Entropy's Strategy [Chattopadhyay et al. 2012] and initialization special improvements [Stetco et al. 2015].

Each iteration of Fuzzy C-means minimizes the Equation (3) where *N* is maximum number data, *K* is maximum number groups, μ_{ij} is element pertinence of example *i* and group*j*, x_i is the example of *i* position in data vector *(X)* and c_j is *j* group centroid.

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{K} \mu_{ij}^m ||x_i - c_j||^2. \quad (3)$$

The sum of one example membership with *n* groups should be 1. The individual value of each pertinence in one group should have interval between 0 and 1. C-means randomly initializes the pertinence matrix μ.

Until the stop criterion is not satisfied, the algorithm keeps calculating centroids and membership matrix. The input is a data vector and the intervals of the groups. The output is membership matrix considering all data. The centroids of the groups (c_j) are determined by Equation (4).

$$c_j = \frac{\sum_{i=1}^{N} \mu_{ij}^m . x_i}{\sum_{i=1}^{N} \mu_{ij}^m}. \quad (4)$$

The update of pertinences should have only number less or equal 1. Furthermore, in Equation (5), the division is 1 for the sum of inter-clusters powering by a division that represents the size of algorithm change. Fuzziness' coefficient (m) is responsible for the velocity and level of details. Smaller m represents a slow and detailed algorithm. The value of m should be carefully chosen, since a big value may not allow the algorithm to find good results.

$$\mu_{ij}^m = \frac{1}{\sum_{k=1}^{C} \frac{||x_i - c_j||}{||x_i - c_k||}^{\frac{2}{m-1}}}. \quad (5)$$

5. Metrics to choose clusters' number

5.1. Davies-Bouldin

The separation metrics proposed by Davies and Bouldin (Davies-Bouldin, db) calculates the relationship of dissimilarities inter-clusters and intra-clusters. This metrics tries to find which number of clusters have the best combination between groups [Vesanto et al. 2000] [Davies and Bouldin 1979]. Thus, it has to accomplish these following rules: (i.) Similarity should be non negative; (ii.) The symmetry property should be true; (iii.) When the distance inter-clusters rises without dispersion change, the similarity inter-clusters decreases; (iv.) When dispersion of clusters rises without distance inter-clusters change, the similarity inter-clusters rises.

Therefore, the dissimilarity inter-clusters is calculated between centroids' distance in Equation (6). Inside a group, the dissimilarity is calculated as the Equation (7), in other words, it is calculated as the sum of all distances between each point (x_i) and centroid (c_r) divided by the total number of elements inside each group (r). The math expression $||.||$ represents the Euclidean Norm by two elements in their dimensions.

$$d_{ce}(l, r) = ||c_r - c_l||, \tag{6}$$

$$S_c(r) = \frac{\sum_i ||x_i - c_r||}{N_r}. \tag{7}$$

The relation between dissimilarities is seen in Equation (8) where the maximization of Equation (9) is chosen. Thus, the lower value of number of groups for the farthest clusters is the answer.

$$db = \frac{1}{R} \sum_{r=1,i=1}^{R} \max(db(l, r)), \tag{8}$$

$$db(l, r) = \frac{S_c(r) + S_c(l)}{d_{ce}(l, r)}. \tag{9}$$

5.2. Gap Statistic

Gap Statistic is classified as an efficient metrics to find the best number of groups for a clustering task. This metrics was compared with Hartigan [Hartigan 1975], KL [Krzanowski and Lai 1988], CH [Calinski and Harabasz 1974] and Silhouette [Kaufman and Rousseeuw 1990]. The majority of tests realized by Tibshirani, Walther e Hastie, *Gap* achieved the best results [Tibshirani et al. 2001].

The database used is dense and has high probability of overlapped groups. Gap Statistic is appointed as the best metric to choose number of groups for Clustering Algorithms [Tibshirani et al. 2001].

Equation (10) calculates the sum of distances ($d_{ii'}$) between two points in positions i and i' for each r group. Equation (11) sum all distances divided each one for the double of distances' quantity for each group *($2n_r$)*. Equation (12) compares expected value of random sample (E_n^*) using database limit. The logarithm is just used to diminish behaviour values. Therefore, the best value of groups should be minor number k that the

mathematical expression is true $Gap(k) \geq Gap(k+1) - s_{k+1}$ where s_{k+1} is standard deviation in $k+1$ group.

$$D_r = \sum_{i,i' \epsilon C_r} d_{ii'}, \tag{10}$$

$$W_k = \sum_{r=1}^{R} \frac{1}{2n_r} D_r, \tag{11}$$

$$Gap_n(k) = E_n^*(log(W_k)) - log(W_k). \tag{12}$$

6. Experiments and Results

The first step of KDD is Data Selection. We chose two datasets for the experiment. One set is named as G947 because have 947 students that interacted at least answering 16 questions of each subject in the functionality selected. The second one is named as G3241 because it is formed by 3241 users that answered 8 questions of each subject in the functionality selected. 16 and 8 values were picked out because are the maximum and medium number of questions solved by the students of the database.

The second step is Data Preprocessing. The database has not demonstrated any discrepant, absent or chronological error. The third step is Data Transformation. It was necessary to normalize the percentage of students that correctly answers and the percentage of answered questions for each subject. Each question information is represented in the platform as a query or affirmative that can be answered by true or false or by a choice of five different alternatives.

This section is divided in three subsections. The Subsection 6.1 uses a group of 947 users in a educational platform to detect which algorithm has a better result, K-means or C-means, and which number of subjects has less uncertainty, 11 or 17. In the Subsection 6.2 we analyze the clustering with 947 and 3241 users. In Subsection 6.3, the relation of subjects and users are explained using both experiments.

6.1. Algorithms' Convergence and Number of subjects

The sample utilized to analyze the difference in the convergence between K-means and C-means and the number of subjects was formed by all the users that answered at least the maximum quantity of subject's question that has fewer questions in the platform. This group G947 has students that interact a lot with the platform. In Figure 1, it is possible to compare the difference between inter-cluster and intra-cluster dissimilarity. The inter-cluster should be the biggest value to choose, and the intra-cluster should be the smallest value to choose. This means that the difference of a big number with the small number should have a greater number. In Figure 1, it is possible to notice that C-means has higher values. This means that C-means is better than K-means. In Figure 2, it is possible to notice the same thing. Independent of subject's number, C-means has higher value of the dissimilarity's difference than K-means.

We used the Shapiro-Wilk [Shapiro and Wilk 1965] test and we concludes that the distribution of the results does not follow a Normal Distribution. Thereafter, we selected the signed-rank Wilcoxon Test [Wilcoxon 1945] [LITCHFIELD and Wilcoxon 1949] to prove the inequality of K-means and C-means convergence. Hence, both algorithms are proved different and the *C-means* approach presented better results.

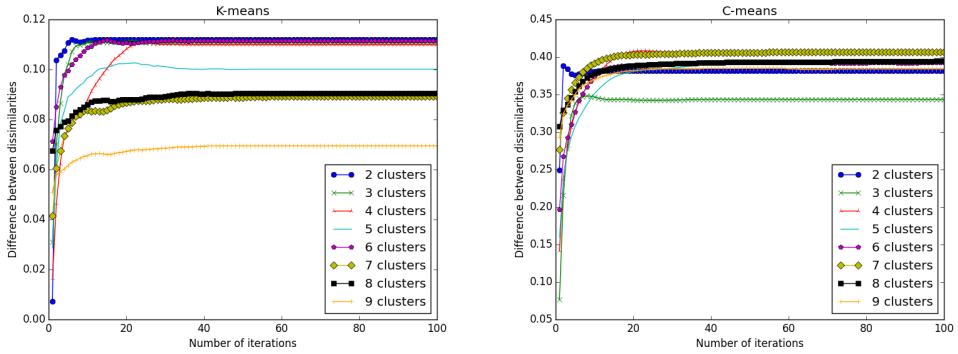

Figure 1. Difference between inter-cluster and intra-cluster dissimilarities in K-means and C-means algorithms using G947-17 set.

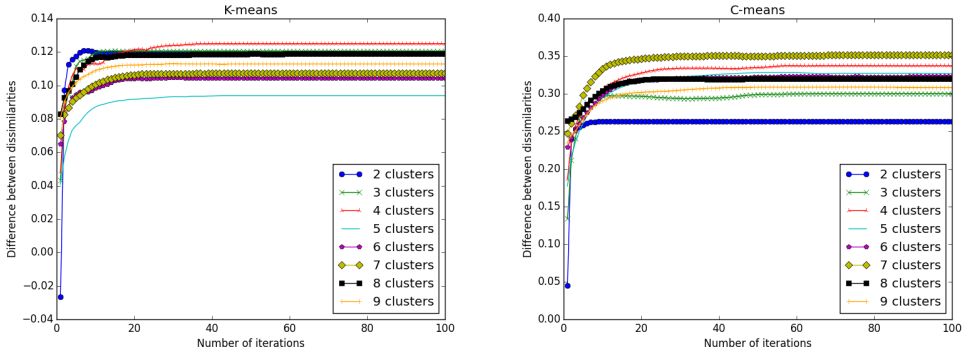

Figure 2. Difference between inter-cluster and intra-cluster dissimilarities in K-means and C-means algorithms using G947-11 set.

In Tables 1 and 2, the symbol N_g means the number of groups, *Dis* means type of dissimilarity (inter-cluster or intra-cluster), *Mean* is the mean of dissimilarity's results and *StdDev* is the value of standard deviation of dissimilarity's results. Comparing Table 1 and 2, one can observe that using 11 subjects leads to lower values of standard deviation, thus less uncertainty. Moreover, we noticed that using 11 subjects, the system finds users with more similarity inside the clusters.

6.2. Number of clusters

In this subsection, we used both sets G947 and G3241. C-means is used because it generated better results than K-means. Thus, the configuration of sets' dimensions is going to be used by 11 subjects. Moreover, the configuration of 22 dimensions is added to both sets since in the case with 11, the number of questions' answered by each user is not analysed. The 22 dimensions are the combination of 11 percentages' efficient in each subject and 11 percentages' answered question quantities.

Experiments'result of **G947-11** e **G947-22** are demonstrated by Figure 3 and 4. For the Davies-Bouldin Metrics, it is minimized for two clusters. The same happened to the Gap Statistic.

Experiments for **G3241-11** and **G3241-22** are demonstrated in Figures 5 and 6. Davies-Bouldin Metric is minimized considering standard deviation using both of config-

Table 1. Table of dissimilarities' values of C-means algorithm using G947-17 set.

N_g	Dis	Mean	StdDev
2	Inter	0.7526	7.49e-16
3	Inter	0.7808	0.044964
4	Inter	0.8241	0.054161
5	Inter	0.8011	0.092341
6	Inter	0.7929	0.113310
7	Inter	0.8026	0.145801
8	Inter	0.7859	0.125607
9	Inter	0.7717	0.104099
2	Intra	0.4719	2.75e-16
3	Intra	0.4371	0.006237
4	Intra	0.4178	0.005668
5	Intra	0.4065	0.002396
6	Intra	0.4007	0.004061
7	Intra	0.3955	0.004296
8	Intra	0.3912	0.003236
9	Intra	0.3873	0.002130

Table 2. Table of dissimilarities' values of C-means algorithm using G947-11 set.

N_g	Dis	Mean	StdDev
2	Inter	0.6343	8.05e-16
3	Inter	0.6416	0.040535
4	Inter	0.6637	0.036251
5	Inter	0.6431	0.050978
6	Inter	0.6314	0.055261
7	Inter	0.6554	0.095027
8	Inter	0.6182	0.076394
9	Inter	0.6015	0.055523
2	Intra	0.3711	1.58e-16
3	Intra	0.3414	0.005332
4	Intra	0.3264	0.003561
5	Intra	0.3160	0.002232
6	Intra	0.3084	0.003267
7	Intra	0.3037	0.003941
8	Intra	0.2980	0.002852
9	Intra	0.2931	0.002159

uration for 2 clusters. The same happened to the Gap Statistic.

Independent of users' number using the same strategy, we found the best results for 2 clusters. This occurs because the users are very different considering their characteristics. Moreover, it is important to reveal that the answer of both metrics was consistent because of the pre-processing stage. The use of not refined data caused divergence in preliminary results.

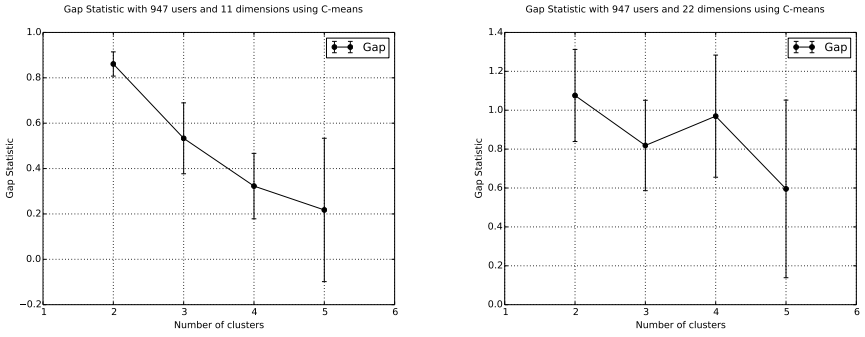

Figure 3. Gap Statistic graphic using C-means in G947-11 and G947-22.

6.3. Users' profile

The users' profiles were divided into two groups. Table 3 shows the profile 1 with more high correlations than 0's profile using Spearman's rank correlation coefficient. Moreover, 1's profile has highest scores than profile 0. Profile 1 represents students

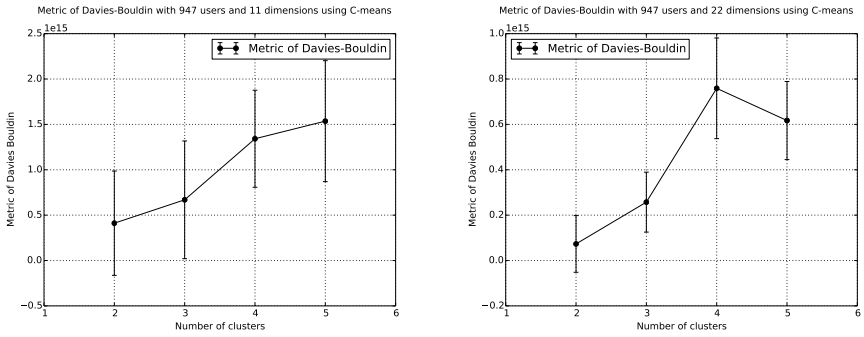

Figure 4. Davies-Bouldin graphic using C-means in G947-11 and G947-22.

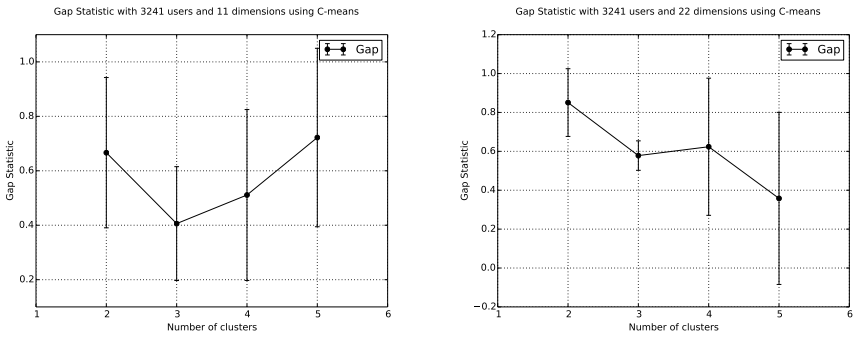

Figure 5. Gap Statistic graphic using C-means in G3241-11 and G3241-22.

who know more about subjects' contents. The group has the highest similarity, **G947** and shows more capability of correlating subjects.

Using Spearman's rank correlation coefficient between profiles, it was proved that profile 0 and profile 1 are different clusters. Thus, C-means could divide the data into two very distinct groups but with small intra-cluster dissimilarity. The 1's profile could lead to important subjects' relations. Moreover, it is possible to search for more similarity in others functionality to know more about the educational platform's users.

7. Conclusions

Educational platforms have been improving their functionalities to identify and understand several problems. Some problems have to be highlighted as how to understand the users and how to identify users' needs. In this paper, we showed that it is possible to answer those questions for a confidential dataset with more than 300 thousand students and 17 subjects.

Considering that the main goal was to find similarities and dissimilarities of different levels of students, we analyzed the unique mensurable functionality, questions' resolution, to divide all the students in profiles. Moreover, it was expected that these profiles emerge peculiar characteristics of using the others functionalities.

We took some decision for the Knowledge Discovery process in Databases. One of KDD's important iteration is to select the set of users. Another aspect relies on how

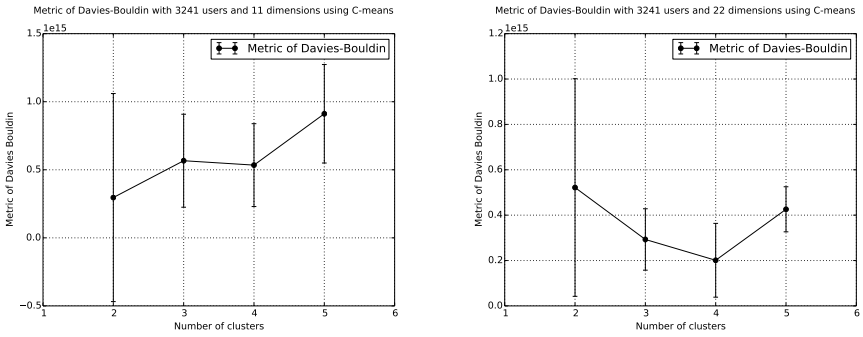

Figure 6. Davies-Bouldin graphic using C-means in G3241-11 and G3241-22.

Table 3. Table with quantities of each interval of values in Profiles 0 and 1 using Spearman's rank correlation coefficient.

Experiment	Correlation Interval	Profile 0	Profile 1
G947-11	$[0.00, 0.25]$	63	10
G947-11	$[0.25, 0.40]$	71	119
G947-11	$[0.40, 1.00]$	2	7
G947-22	$[0.00, 0.25]$	65	13
G947-22	$[0.25, 0.40]$	69	117
G947-22	$[0.40, 1.00]$	2	6
G3241-11	$[0.00, 0.25]$	97	49
G3241-11	$[0.25, 0.40]$	39	87
G3241-11	$[0.40, 1.00]$	0	2
G3241-22	$[0.00, 0.25]$	93	70
G3241-22	$[0.25, 0.40]$	39	64
G3241-22	$[0.40, 1.00]$	0	2

many subjects can be used. K-means demonstrated different convergence values of C-means proved by Shapiro-Wilk and Wilcoxon Test. We used the highest number of subject to achieve the best C-means convergence. Davies-Bouldin and Gap Statistic were important to determine the adequate number of profiles. C-means clustering pointed out two different profiles using Spearman's rank correlation coefficient.

The use of just one functionality found similarities and correlations considering the students that have a higher probability to pass in exams using the platform. Moreover, the found profiles lead to some specific behavior on other functionalities. For future works, it is important to investigate the obtained profiles to discover for information inside them. Each profile can be clustered again.

References

Abdous, M. . H., Yen, W. ., and J., C. (2012). Using Data Mining for Predicting Relationships between Online Question Theme and Final Grade. *Educational Technology and Society, n*, 15(3):77–88.

Bezdek, J. C. (1981). *"Pattern Recognition with Fuzzy Objective Function Algorithms"*. New York: Plenum Press.

Calinski, T. and Harabasz, J. (1974). *A dendrite method for cluster analysis*. Communications in Statistics-Theory and Methods.

Chattopadhyay, S., Pratihar, D., and Sarkar, S. (2012). A Comparative Study of Fuzzy C-Means Algorithm and Entropy-Based Fuzzy Clustering Algorithms. *Computing and Informatics*, 30:701–720.

Davies, D. L. and Bouldin, D. W. (1979). "a cluster separation measure. *" IEEE Trans. Patt. Anal*, PAMI-1:224–227.

Dominguez, A. K., Yacef, K., and Curran, J. R. (2010). Data mining for individualised hints in elearning. In *Proceedings of the 3rd international conference on educational data mining*, page 91–100.

Forgy, E. (1965). Cluster analysis of multivariate data: efficiency vs. *interpretability of classification, Biometrics*, 21(768.).

Full, W., Ehrlich, R., and Bezdek, J. (1984). *FCM : The Fuzzy C-means clustering algorithm*. 10(2):191–203.

Ghosh, S. and Dubey, S. K. (2013). Comparative Analysis of K-Means and Fuzzy C-Means Algorithms. 4(4):35–39.

Gong, Y., Beck, J. E., and Heffernan, N. T. (2010). Using multiple Dirichlet distributions to improve parameter plausibility. In *Proceedings of the 3rd international conference on educational data mining*, page 61–70.

Hartigan, J. (1975). *Clustering algorithms*. John Wiley and Sons, Inc.

Hartigan, J. A. and Wong, M. A. (1979). *Algorithm AS 136 A K- Means Clustering Algorithm, Applied Statistics, 28(1)*. 100-108.

Huei-Tse, H. (2011). A case study of online instructional collaborative discussion activities for problem-solving using situated scenarios: an examination of content and behavior cluster analysis. *Computers Education*, 56(3).

Kardan, S. and Conati, C. (2011). A framework for capturing distinguishing user interaction behaviours in novel interfaces. In *Proceedings of the 4th international conference on educational data mining*, page 159–168.

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, Canada.

Krzanowski, W. and Lai, Y. (1988). *A criterion for determining the number of groups in a data set using sum-of-squares clustering*. Biometrics.

Lan, X., Li, Q., and Y., Z. (2015). Density K-means A New Algorithm for Centers Initialization for K-means.

LEE, Y. (2012). Developing an efficient computational method that estimates the ability of students In: a Web-based learning environment. *Computers and Education. v*, 58:579–589.

Lin, P.-l. (2016). A Validity Index Method for Clusters with Different Degrees of Dispersion and Overlap. pages 222–229.

LITCHFIELD, J. a. and Wilcoxon, F. (1949). A simplified method of evaluating dose-effect experiments. *Journal of pharmacology and experimental therapeutics*, 96(2):99–113.

Lloyd, S. P. (1982). *Least squares quantization in PCM, IEEE Transactions on Information Theory*.

MacQueen, J. (1967). *Some methods for classification and analysis of multivariate observations, Proceedings of the Fifth Berkeley Symposium On Mathematical Statistics and Probabilities*. 1.

Malmberg, J., Järvenoja, H., and Järvela, S. (2013). *Patterns in elementary school student´ strategic actions in varying learning situations*. Instructional Science.

Nugent, R., Dean, N., and Ayers, E. (2010). Skill set profile clustering: the empty k-means algorithm with automatic specification of starting cluster centers. In *Proceedings of the 3rd international conference on educational data mining*, page 151–160.

Peña, J. M., Lozano, J. A., and Larrañaga, P. (1999). *An empirical comparison of four initialization methods for the K -Means algorithm*. 20.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, pages 591–611.

Sparks, R. L., Patton, J., and Ganschow, L. (2012). Profiles of more and less successful L2 learners: a cluster analysis study. *Learning and Individual Differences*, 22(4).

Stetco, A., Zeng, X.-j., and Keane, J. (2015). Expert Systems with Applications Fuzzy C-means ++ : Fuzzy C-means with effective seeding initialization. *Expert Systems With Applications*, 42(21):7541–7548.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *J. R. Statist. Soc. B*, 63(2).

Vesanto, J., Alhoniemi, E., and Member, S. (2000). Clustering of the Self-Organizing Map. 11(3):586–600.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6):80–83.

Yong, Y., Chongxun, Z., and Pan, L. (2004). "a novel fuzzy c-means clustering algorithm for image thresholding", measurement science review. 4(1).