# Application and Comparison of Clustering Methods to Educational Process Data

Meredith Luo

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Education

University of Washington

2022

Committee:

Min Li

Elizabeth A. Sanders

Oscar L. Olvera Astivia

Program Authorized to Offer Degree:

College of Education

University of Washington

## Abstract

Application and Comparison of Clustering Methods to Educational Process Data

Meredith Luo

Chair of the Supervisory Committee:
Min Li
Elizabeth A. Sanders
Oscar L. Olvera Astivia
College of Education

Cluster analysis has great potential for analyzing the vast amounts of process data that record the online learning behaviors of students. It can be used to develop profiles of student groups that help instructors understand students' online learning patterns. However, one of the major challenges in employing cluster analysis is to select a suitable one among many clustering algorithms. This methodological paper introduces and compares three clustering algorithms, including two popular non-hierarchical clustering methods, *k*-means and *k*-medoids, and one hierarchical method called agglomerative hierarchical clustering analysis (HCA). The dataset used for demonstration is a publicly available dataset, Open University Learning Analytics (OULA), which contains information on online modules, student demographics, and students' clicks on the virtual learning environment (VLE). To examine the utility of process features and performance of the selected clustering algorithms in predicting students' module outcome (i.e.,

pass or fail), one module was selected ($N$ = 1299), and 18 process features were developed. After obtaining the clustering results from each algorithm, logistic regression was used to compare and validate the cluster memberships with students' module outcomes (i.e., pass or fail). Multiple logistic regression was employed to explore the demographics and process feature compositions of the most predictive clustering results. The results of the present study showed that *k*-means and *k*-medoids generated comparable results, while agglomerative HCA produced the most dissimilar yet most predictive results compared to *k*-means and *k*-medoids. Multiple logistic regression results showed that students who engaged in certain VLE activities such as taking quizzes or joining discussion forums had a higher chance of being in the high-performance group (i.e., the group with a higher probability of passing the module). Limitations and future research directions were discussed.

## Introduction

To search for the structure among the unlabeled observations based on a multivariate profile, cluster analysis is one of the most commonly used techniques for this purpose. Clustering methods group the unit of analysis (e.g., people, schools, etc.) into clusters in which the objects are similar and dissimilar to those belonging to other groups. Thus, the principle of clustering methods is to maximize the homogeneity of objects within clusters (i.e., intra-cluster similarity) while maximizing the heterogeneity between clusters (i.e., inter-cluster dissimilarity) (Hiar, Black, Babin, and Anderson, 2019). Clustering methods are frequently used in education research. Previous research has found that they were helpful in creating clusters of students that shared similar learning patterns and performances (Dutt, Aghabozrgi, Ismail, & Mahroeian, 2015). With the advancement in online learning platforms, researchers now have access to rich information about students' thinking and learning processes. Clustering methods were able to

leverage that information and provide instructors and students insights into what learning patterns are advantageous. For instance, Wook and colleagues (2009) have applied clustering methods to students' online reading annotations (e.g., highlighting texts or adding notes) and found that a cluster of students with a similar reading style improved reading comprehension faster. However, one of the major challenges in employing clustering methods is selecting an appropriate and informative one among many possible options.

Many clustering methods have been developed, and they can be divided into two major categories: non-hierarchical and hierarchical clustering methods (Gülagiz & Sahin, 2017; Rokach & Maimon, 2005). For non-hierarchical clustering, an initial number of groups ($k$) is prespecified by the researchers. Then, $k$ centroids are randomly placed in a multidimensional space. In an iterative manner, the centroids of each cluster are updated to make the points in a given cluster closer to their centroid, while moving them away from the centroids of other clusters. Centroid and cluster assignments are updated until the cluster assignment stops changing (i.e., until convergence is achieved) (Likas, Vlassis, & Verbeek, 2003). Commonly used non-hierarchical clustering methods include $k$-means clustering (MacQueen, 1967) and $k$-medoids clustering (Rousseeuw & Van Zomeren, 1990).

In contrast to non-hierarchical clustering, hierarchical clustering or hierarchical cluster analysis (HCA) does not require pre-specifying the number of clusters. HCA builds a nested series of partitions using a tree-based representation of the objects, known as a dendrogram (Abbas, 2008). The number of clusters depends on where to cut the dendrogram. HCA has two types, one is agglomerative, and the other one is divisive clustering (Rokach & Maimon, 2005). Agglomerative HCA uses a "bottom-up" approach. Each data is considered a cluster of its own at the beginning. At each step of this algorithm, clusters that are the most similar are merged into a

new cluster. This process stops when all observations are members of one single big cluster at the top (Hiar et al., 2019). Divisive clustering is the inverse of the agglomerative method and is less commonly used (Reddy, Makara, & Satish, 2017).

These two categories of clustering methods have their own merits and disadvantages. Past research has found that clustering methods similar to *k*-means have relatively low time complexity and high computational efficiency. However, they are sensitive to outliers, and their results are greatly determined by the number of prespecified clusters (Kaushik & Mathur, 2014). Hierarchical clustering methods have higher time complexity and assume nestedness among clusters, but they are more robust to outliers (Xu & Tian, 2015). Besides the specific features for each category, the clustering methods mentioned have other important characteristics in general. The same clustering algorithm can produce different results depending on the types of similarity measures employed (Yim & Ramdeen, 2015). Additionally, clustering methods will always create clusters, regardless of the actual existence of any structure in the data. Each observation is deterministically assigned to one and only one specific cluster. Therefore, the clustering results can be inconsistent and arbitrary in themselves (Reddy et al., 2017). In the scope of the present study, it was of methodological interest to compare and validate the performance of *k*-means, *k*-medoids, and agglomerative clustering algorithms.

To apply and validate the selected clustering algorithms, this study used a well-known publicly available dataset called Open University Learning Analytics (OULA), which the Open University provides (Kuzilek, Hlosta, & Zdrahal, 2017). OULA contains information about online courses, demographic information of students, and students' interactions with the Virtual Learning Environment (VLE) for each course. The primary goal of past studies on the OULA dataset was to predict students' outcomes accurately (e.g., pass or fail) based on student

demographics and VLE process features (e.g., number of clicks for web pages). To achieve this purpose, most of the studies explored various supervised machine learning models using the information of all students across different courses altogether (Hlosta, Papathoma, & Herodotou, 2020; Azizah, Pujianto, & Nugraha, 2018; Heuer & Breiter, 2018; Hussain, Zhu, Zhang, & Abidi, 2018). For example, Azizah and colleagues (2018) used geographic region, education level, disability condition, number of web pages viewed, etc., of all students to predict students' course outcomes (i.e., passing or failing). They built supervised machine learning models, such as Naive Bayes, and obtained each model's prediction accuracy rate. No studies up to date have compared different clustering algorithms' performance on this dataset, and none has tried to predict student outcomes solely based on the VLE process features. Therefore, to address this gap in the literature, the contribution of the present study is twofold: 1) to compare and validate the clustering algorithms with the students' outcomes, and 2) to examine the utility of process data in its own right to predict student course success.

### Research Goals

Using the OULA dataset, the present study focuses on achieving the following three research goals:

1. Compare three popular clustering methods, *k*-means, *k*-medoids, and agglomerative HCA, to partition students into groups based on their online learning behaviors.
2. Validate the clustering results with students' final course results.
3. Profile the clusters using multiple logistic regression analysis.

## Methods

**Dataset**

The OULA dataset consists of several files, including 32,593 students' demographic, VLE description, VLE process data represented by the sum of clicks in a day, assessment scores, and course information. The seven courses are called modules, consisting of three social science and four STEM modules, which were presented either in October or February 2013 or 2014. The VLE data is comprised of different types of activities, and each represents an important element of the VLE. Students interacted with these elements to access course material, participate in discussion events, or take quizzes. The final course results of students are classified into four categories: distinction ($n$ = 3,024), pass ($n$ = 12,361), fail ($n$ = 7,052), and withdrawn ($n$ = 10,156).

The present study focused on exploring the usefulness of the VLE process data in clustering students. Therefore, this study removed withdrawn students since many of them had few or no records of VLE activity. Additionally, students could attempt the same module more than one time. In this case, the process activities for students who attempted multiple times could be different from those of the first attempts. These differences might not be due to the academic performance variation among students, which is what the current study intends to investigate, but due to students' previous experiences with the course material. Therefore, to remove this potential confounding effect, only the process data of the first attempt remained. After removing the withdrawn and multiple attempts students, only one module offered at a specific month was selected instead of utilizing all the data across different modules. This step ensured that every individual data point in each process feature variable had the same contextual interpretations. Finally, a STEM module, referred to as FFF, offered in October 2014, was chosen because

compared to other modules, it has a larger sample size ($n$ = 1299) and a more reasonable proportion of passing and failing (78% pass or distinction, 22% fail).

**Variables**

There were 18 variables generated from students' VLE activities in total (See Table 1). The first 15 variables were about the 15 types of VLE activities within the selected module. For each student, the sum of clicks for each type of activity was computed. Besides the 15 variables, the duration of taking the module for each student was computed. It was defined as the number of days from the first day of the VLE activity to the last day of the VLE activity. Another variable was the number of active days, which was the number of days the students had any interaction with the VLE. (0.8) The last variable was the first assessment score for each student. Hlosta and colleagues (2017) found that based on the information of the first assessment, they had a decent F1 measure (71.31%) in identifying at-risk students. Thus, the first assessment score, which might be an essential factor in clustering students, was included in this study.

**Clustering Analysis**

**Determine the number of clusters.** *K*-means and *k*-medoids algorithms need to prespecify the number of clusters (*k*). The current study used two approaches to choose *k*. The first approach uses the silhouette width, introduced by Kaufman and Rousseeuw (2009). The concept of silhouette width involves the difference between the within-cluster tightness and the inter-cluster separation. Specifically, the silhouette width *s(i)* for observation *i* is defined as:

$$s(i) = \frac{b(i) - a(i)}{max(a(i), b(i))} \quad (1)$$

Where *a(i)* is the average distance between *i* and other members in the same cluster. *b(i)* is the minimum of the average distances between *i* and all the other objects. The values of the

silhouette width lie in the range from -1 to 1. If the silhouette width is close to -1, then it means that the object is misclassified; values near 0 mean that the object is equally likely to be in this group or in other groups; If the average value for a cluster is close to 1, the cluster is well clustered. Thus, the goal is to find which *k* gives the largest average silhouette width when plugging multiple *k* values into the *k*-means and *k*-medoids algorithms.

Another approach is to use the resampling method and examine cluster stability (Hennig, 2007). Stability means a meaningful, valid cluster should not disappear easily if the data set only went through non-essential changes (e.g., resampling via bootstrap). Therefore, the goal is to find under which *k* the clusters are the most stable, in other words, the most meaningful and valid. Specifically, this approach resamples the data via non-parametric bootstrap with replacement, computes the similarity of the resulting partitions for each *k*, and selects the *k* where the similarity measure is the highest. Jaccard coefficient (Jaccard, 1901) was used to measure the closeness of the partition results (see equation 2).

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

A is the cluster generated from the original dataset, and B is the cluster yielded from a bootstrapped sample. The Jaccard coefficient gives the proportion of objects belonging to both clusters of all the objects involved in at least one of the clusters. By design, *J(A, B)* ranges from 0 to 1. For example, if the data was bootstrapped 100 times and for each bootstrapped sample, *k*-means with a *k* set to two was performed, then there are 200 Jaccard coefficients computed. The average of 200 Jaccard coefficients can represent cluster stability for *k*-means clustering when *k* equals two. According to Hennig (2008), a valid and stable cluster should yield a mean Jaccard

coefficient of 0.75 or more. Below 0.6, clusters should not be trusted. If the value is above or equal to 0.85, the clusters are considered highly stable.

***K*-Means.** The goal of *k*-means is to minimize the total within-cluster variation. The standard procedure used is the Hartigan-Wong algorithm (Hartigan & Wong, 1979). It defines the total within-cluster variation as the sum of all of the pairwise squared Euclidean distances between the observations in the clusters. Equation (3) defines the optimization problem of *k*-means clustering.

$$\underset{\text{minimize}}{C1, \ldots C_K} \left\{ \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2 \right\} \quad (3)$$

where *p* is the total number of features included in the model. $x_{ij}$ is the observation *i*'s value for feature *j* in cluster $C_k$. xkj is the mean for feature *j* in cluster $C_k$.

There are three major steps in the *k*-means algorithm:

1. Specify the number of clusters K
2. Randomly partition *n* observations into K clusters
3. Iterate the following two steps until the cluster assignments stop changing:
    0. For each cluster, compute the cluster centroid. The centroid is a vector of *p* feature means for the observations in each cluster.
    1. Assign each data point to the cluster whose centroid is the closet, which is defined using the Euclidean distance.

***K*-Medoids.** *K*-medoids is very similar to the *k*-means algorithms. One of the most salient differences is that *k*-medoids chooses actual data points as centers, whereas *k*-means uses the average between the points as the centers. Because *k*-medoids is based on the most centrally located object in the cluster, it is less sensitive to outliers compared to *k*-means. Moreover, *k*-

medoids can use similarity measures other than Euclidean distance, while *k*-means generally require it for efficient solutions. The current study used Manhattan distance, which is the sum of absolute distances rather than the root of sum-of-squares differences in Euclidean distance.

The *k*-medoids algorithm used in this paper was introduced by Park and Jun (2009). There are three major steps in the *k*-medoids algorithm after specifying the *k*:

1. Select initial medoids:
    a. Calculate the distance (d) between every pair of all objects based on Manhattan distances
    b. Calculate vj for object j (in total of n objects) as follows:

$$v_j = \sum_{i=1}^{n} \frac{d_{ij}}{\sum_{l=1}^{n} d_{il}}, j = 1, ..., n \quad (4)$$

    c. Sort the vj in ascending order and select k objects having the first k smallest values as initial medoids
    d. Assign each object to the nearest medoid and calculate the sum of distances from all objects to their medoids
2. Update medoids: Find a new medoid for each cluster to minimize the total distance to other objects in its cluster.
3. Assign objects to medoids:
    a. Assign each object to the nearest medoid
    b. Calculate the sum of the distances from all objects to their medoids. Iterate until the sum of the distances is equal to the previous one. Otherwise, repeat steps 2 and 3.

**Agglomerative HCA.** It has two major steps (Bouguettaya et al., 2015).

1. Treat each observation as its own cluster. Select a dissimilarity measure to calculate the dissimilarity measures of all pairs of clusters. In this study, the Euclidean distances of all pairs were used as the dissimilarity measure.
2. Iterate until there is only one cluster left:
    a. Examine all pairwise inter-cluster dissimilarities using Ward's minimum-variance method and fuse two clusters that are least dissimilar. In Ward's minimum-variance method, the distance between two clusters is the ANOVA sum of squares between the two clusters added up over all the variables. At each generation, the within-cluster sum of squares is minimized over all partitions obtainable by merging two clusters from the previous generation (Ward Jr, 1963). The values of the Ward measure represent the height of the dendrogram at which merging occurred.
    b. Calculate the new pairwise inter-cluster dissimilarities among the remaining clusters.

**Cluster Comparison and Validation**

Logistic regression was employed to predict students' outcomes based on the clustering results from each clustering algorithm. The aim was to evaluate how predictive each clustering result was and determine which algorithm produced the most predictive cluster membership. The dependent variable, module outcomes, was dummy coded (pass or distinction as 1 and fail as 0). Three separate logistic regression models were built for each clustering algorithm.

$$\text{Logit (Pass Module)} = b_0 + b_1 * \text{K-Means\_Result}$$

$$\text{Logit (Pass Module)} = b_0 + b_1 * \text{K-Medoids\_Result}$$

$$\text{Logit (Pass Module)} = b_0 + b_1 * \text{HCA\_Result}$$

The log-odds (logits) of passing the module are equal to the conditional mean ($b_0$), plus the unique effect of cluster membership from each algorithm ($b_1$). Model fit indices such as BIC and

Pseudo $R^2$ values for each of the models were computed and compared to see which model had the most predictive cluster membership.

### Cluster Profiles

After determining the algorithm with the most predictive cluster result, multiple logistic regression was used to profile its produced clusters. The dependent variable was the cluster membership of the students. Three models were constructed to understand a) the demographic composition of the groups, b) the process variable composition of the groups, and c) which variables of both demographics and process features contributed to the groups. The demographic variables included gender, geographic regions, age range, and disability condition of each student. For ease of results interpretation, all the demographic variables were effect coded, and all eighteen process variables were standardized.

Logit (Group) ~ Demographics

Logit (Group) ~ Process Features

Logit (Group) ~ Demographics + Process Features

## Result

### Descriptive statistics

Detailed descriptions of eighteen process variables are provided in Table 1; Table 2 shows the descriptive statistics (i.e., mean and standard deviation) and Spearman correlations among all eighteen variables.

### Cluster Analysis Results

**Number of Clusters.** To determine the optimal number of clusters for *k*-means and *k*-medoids, silhouette width values and Jaccard coefficients over different numbers of clusters *k* were plotted. Figure 1 indicated that the highest silhouette width value for both algorithms occurs when there are two clusters. Silhouette width suggests how well the objects fit into the assigned clusters. A larger silhouette width value (i.e., closer to 1) means the observation is well placed in its cluster. Thus, when the number of clusters was two, the observations were best clustered under both algorithms. The averaged Jaccard coefficients over different numbers of clusters further confirmed the finding (Figure 2). The two-cluster structure gave the highest Jaccard coefficients, which were above 0.9 for both algorithms. Thus, the clustering results obtained via bootstrapping were very similar and stable when the number of clusters was two. In short, these two measures agreed and implied that there were two meaningful and valid clusters in this dataset. Hence, *k* was set to two for *k*-means and *k*-medoids in the following analysis.

For agglomerative HCA, the number of clusters *k* does not need to be specified. Instead, a dendrogram was plotted to examine the data structure by showing the results of running the observations through agglomerative HCA (Figure 3). The height axis of the dendrogram represents the dissimilarity measures calculated by Ward's minimum variance between clusters. The horizontal bars represent the point at which clusters/observations are merged. Figure 3 reveals two major groups in the dataset. Hence, by observing the dendrogram, this study decided to cut the dendrogram at which the observations were divided into two major clusters.

**Clustering Results.** Figure 4 visualizes the partitioning results for each algorithm. Points in the plot represented observations. The primary and secondary principal components were the x- and y-axis since there were more than two variables. Each dimension represents a certain amount of variation in the original dataset. In this case, dimensions one and two represent 47%

and 8.9%, respectively. An ellipse was drawn around each cluster. On this two-dimensional plot, the two clusters were not well-separated and had some overlapping areas across all algorithms.

Table 3 provides more cluster-specific information for each algorithm, including the number of students, percentage of passing for each cluster, and average silhouette width for each algorithm. *K*-means and *k*-medoids had comparable results, having a similar number of students and a percentage of passing for each cluster. Cluster two had a better passing rate compared to cluster one. Both algorithms had an average silhouette width value of 0.38, indicating that, on average, the observations were reasonably grouped, but the structure was not strongly supported. As for agglomerative HCA, it had fewer students classified as cluster one and included more students in cluster two. Still, the passing rate for cluster two remained high, plus having a lower passing rate for cluster one. It suggested that the clusters generated by agglomerative HCA were in more accordance with the passing outcome. Its slightly higher average silhouette width (0.40) further supported that its clusters were better separated compared to those of the other two algorithms.

**Cluster Comparison and Validation**

Three logistic regression models with each algorithm's cluster membership were used to predict module outcome using a sample of $N = 1299$ students who took the module (Table 4). According to the chi-square test results, the cluster memberships obtained from all algorithms were found to have a significantly better model fit than the null model with no predictors ($p < 0.001$). This indicated that all three kinds of cluster membership help reliably distinguish between students who passed the module from those who did not. The variance in *k*-means cluster membership accounted for was 0.11 using McFadden's (1974) pseudo R2, and K-medoids had a slightly higher percentage, which was 0.16. The highest value belonged to the

agglomerative HCA's model, which was about 0.25. McFadden (1977)stated that values of 0.2 to 0.4 represent an excellent fit. Therefore, cluster membership from agglomerative HCA had the highest predictive value for predicting whether students passed or failed. The BIC values further support this claim. A lower BIC value is preferred because it indicates that the probability of obtaining the given data is higher if the tested model is proposed. The agglomerative HCA model had the lowest BIC value ($BIC$ = 1027.48), and the differences between its BIC value and the two other models' were greater than 155. In short, based on McFadden's pseudo R2 and BIC values, the cluster membership given by the agglomerative HCA algorithm was the most predictive of the module outcome.

For brevity, only the coefficient estimates from the agglomerative HCA model are interpreted here. Model results showed that the intercept was not significantly different from zero. In other words, the mean predicted probability was not significantly different from 50%. The log-odds of passing the module across the sample (holding the cluster membership constant) was $b$ = 0.05 ($SE$ = 0.09), $p$ = 0.597 (mean predicted probability of passing was 51%). Cluster membership was uniquely predictive of module outcome ($b$ = 2.96, ($SE$ = 0.19), $p < 0.001$, $OR$ = 19.25). The $OR$ indicates that students in cluster two (i.e., dummy coded as one) are, on average, 19.25 times more likely to pass the module than students in cluster one. Computing the predicted probabilities based on the model estimates provides a clearer interpretation: students from cluster two had a 95% predicted probability of passing the module compared to cluster one students with a 51% predicted probability. In this case, cluster two was classified as the high-performance group, while cluster one was the low-performance group.

**Cluster Profiles Results**

Three multiple logistic regression models were built to examine the demographic and VLE process characteristics of clusters generated by agglomerative HCA (Table 5). Due to the enormous size of the effect sizes in the Odds Ratio metric, this study chose to report the results in the logit metric. Two chi-square tests were performed to compare model one and two with the complete model, model three. It found that model one and three were significantly different $X^2$ ($df$ = 18, $p < 0.001$), meaning that demographics did not explain much of the deviance in cluster memberships. On the other hand, model two and model three were not significantly $X^2$ ($df$ = 16, $p > 0.05$), suggesting model one can explain as much as deviance as the full model. According to the principle of parsimony, which perfers models with fewer predictors, this study will focus on discussing the results from model two. The interpretations of the results for the other models were only presented in the result section.

The first model only included demographic information, and two of the variables were significantly associated with the cluster membership (i.e., gender and age). For gender ($b$ = -0.33, ($SE$ = 0.08), $p < 0.001$), male students had a predicted probability of 65% being in cluster two (i.e., high-performance group), while female students had a predicted probability of 78% to be in the high-performance cluster when controlling for all other demographic variables. For students between age 0 and 35 ($b$ = -0.65, ($SE$ = 0.29), $p < 0.05$), they had a predicted probability of 57% of the high-performance group, while for people between 35-55, the predicted probability was 80%. Thus, based on model one's results, being female and older than 35 boosted one's chance of being in cluster two (the high-performance group).

Model two, including process feature variables, gives insights into which learning behaviors were significantly associated with the cluster groups. The intercept was significantly different from zero, and the predicted probability of being in cluster two was very close to one if

every process activity was on its average level. Among all process features, the sum of the clicks on the contents of an assignment (i.e., oucontent), module quizzes (i.e., quiz), discussion forum (i.e., forumng), and the additional video/audio information (i.e., dataplus) were positively and significantly associated with cluster two membership. This was especially true if a student had a lower than average (-1 SD) sum of clicks on the contents of an assignment ($b$ = 25.36, ($SE$ = 3.65), $p < 0.001$). In this case, the student's predicted probability of belonging to cluster two was close to 0. On the contrary, the sum of clicks on site and activity information (i.e., dualpane), and other sites enabled in the module (i.e., subpage) were negatively and significantly associated with cluster two membership.

Finally, model three contained all demographic and process variables. The intercept of model three was not significantly different from zero, meaning the mean predicted probability was not significantly different from 50%. As for the demographic characteristics, gender and age were no longer significant after adding the process features. The "significant" effect of gender and age observed in the previous model might reflect the effect of process features on cluster membership. Instead, being in the East Anglian ($b$ = -4.19, ($SE$ = 1.71), $p < 0.05$) or West Midlands ($b$ = 2.37, ($SE$ = 0.99), $p < 0.05$) regions had a significant predictive power of cluster membership. The composition of significant process features did not change much after controlling for the demographic variables. The sum of clicks on the information related to the module became positively significantly associated with being in cluster two, and the sum of clicks on other sites enabled in the module had no significant effect. Other than these two differences, the effects of other process variables remained unchanged.

## Discussion

The current study employed three popular clustering algorithms to group students based on the process features obtained from students' online learning behaviors. The findings showed that there were two major student clusters. All three algorithms, *k*-means, *k*-medoids, and agglomerative HCA, delivered similar grouping results. In addition, the module outcomes matched the best with the cluster designations derived from agglomerative HCA. Finally, multiple logistic regression revealed that demographic information, such as region, and process features, such as the sum of clicks on the quiz section, were significantly associated with the cluster membership produced by agglomerative HCA.

The finding that the *k*-means and *k*-medoids yielded similar results was not surprising since they both belong to the non-hierarchical clustering category and possess similar algorithmic steps (Gülagiz & Sahin, 2017). The only difference in their algorithms is that *k*-medoids uses the most centrally-located objects as the centroids for each cluster while *k*-means uses the mean of the coordinates of the objects. This small change helps *k*-medoids become less sensitive to outliers than *k*-means, and past studies have corroborated this advantage (Park & Jun, 2009; Arbin, Suhaimi, Mokhtar, & Othman, 2015; Arora, 2016). In the current study, *k*-medoids did not perform better than *k*-means. Future studies can explore under what circumstances, *k*-medoids' advantage can be evident.

Findings from silhouette width values and logistic regression models agree that the hierarchical clustering method, agglomerative HCA, produces the most different and the most predictive cluster membership for passing or failing the module. However, it is difficult to understand the contributing factors to this result. Previous research comparing HCA and *k*-means rarely focused on cluster validity or accuracy but on time and space complexity (Verma et al., 2012; Karthikeyan, George, Manikandan, & Thomas, 2020; Gupta, Sharma, & Akhtar, 2021).

Most studies only report that *k*-means were faster and used less space than agglomerative HCA. Only Gupta and colleagues (2021) examined the accuracy of *k*-means and agglomerative HCA in clustering diabetes and hypothyroid patients. They agreed with the present study's findings that HCA produced more accurate cluster memberships. Still, results from Gupta and colleagues (2021) and the current paper cannot be used to assert that agglomerative HCA is a better approach under any circumstances compared to *k*-means and *k*-medoids. More investigations need to be conducted to reveal what specific factors contributed to HCA's more accurate cluster results in this study.

Two clusters were formed based on agglomerative HCA. In cluster one, only 51% (N = 516) of students passed the module, while in cluster two 95% (N = 783). Based on this information, cluster two was deemed the high-performance group. According to the results from model two, students who were active in the discussion forum, quiz sections, module assignments, and actively searching for course material were more likely to be in the high-performance group. However, students who clicked more on the sites extra to the module or webpage about activity information were less likely to be in the high-performance group. Knowing this information, instructors can encourage students to be more active in the discussion forums and quiz sections to boost their chances of passing the module.

**Limitations and Future Directions**

The current study has several limitations. First, the model comparison and validation conclusions were limited to this module dataset, meaning that the advantage of agglomerative HCA cannot be generalized to other datasets. However, it is still crucial to investigate what the contributing factors are for HCA to produce the most predictive clustering results in this study because they might be helpful for other researchers to consider when selecting clustering

algorithms. Future research can shed light on the contributing factors to HCA's advantage in the current study by altering the following two aspects. One is to change the selected distance/similarity measures for agglomerative HCA to see if HCA can still produce the most predictive results. Yim and Ramdeen (2015) found that changing the combination of distance and cluster similarity measures in HCA could moderately alter the final cluster results. Thus, it is probable that the measures selected in the current study helped agglomerative HCA obtain accurate results. Another way is to alter the distribution of the variables. Templ, Filzmoser, and Reimann (2008) found that the skewness of the variable distribution influenced clustering results. They found that for hierarchical and non-hierarchical methods, using skewed data with no transformation leads to an increased number of misclassified observations. However, using the Ward method in HCA delivered relatively stable results independent of data transformation. Therefore, potentially, the agglomerative HCA employing the Ward method performs better with skewed variables. Therefore, to test this hypothesis, future research can transform the skewed count variables to examine if HCA can still produce the most accurate results.

Another limitation regards the generation of the process features. The publicly available information about the contents and the interactive interface of the online modules was limited. Hence, it is challenging to develop process variables that are meaningful to instructors and students. Future research can consult instructors or explore relevant theoretical frameworks about online learning behaviors to generate more interpretable and meaningful process variables from students' log data. In this case, the results from the cluster analysis will provide more relevant insights for both instructors and students.

## Conclusion

The contribution of this study is two-fold. First, this study demonstrated the utility of process data in its own right to predict student course success. In contrast to past studies that used all information about students, including their demographic information, the present study performed clustering algorithms using only process features. As a result, the predicted cluster memberships were significantly associated with the module outcomes. This study helped show that students' online behaviors contain rich and valuable information that entails students' learning progress and outcomes. Second, the present study compared and validated three popular clustering algorithms. Past studies had mostly focused on implementing supervised machine learning on this dataset, while the current study examined the performance of clustering algorithms. All three algorithms, *k*-means, *k*-medoids, and agglomerative HCA, generated reliable clustering results predictive of module outcomes. Among the methods, current evidence suggests that agglomerative HCA performed the best. In short, this study demonstrated that it is feasible and promising to employ clustering algorithms to leverage the usefulness of process features, especially with the development of online learning platforms and the increasing availability of log data.

## References

Abbas, O. A. (2008). Comparisons between data clustering algorithms. *Int. Arab J. Inf. Technol.*, *5*(3).

Arbin, N., Suhaimi, N. S., Mokhtar, N. Z., & Othman, Z. (2015). Comparative analysis between k-means and k-medoids for statistical clustering. In *2015 3rd International Conference on Artificial Intelligence, Modelling and Simulation (AIMS)* (pp. 117–121). IEEE.

Arora, P. (2016). Deepali, and S. Varshney,"Analysis of K-means and K-medoids algorithm for big data,." *Procedia Comput. Sci*, *78*, 507–512.

Azizah, E. N., Pujianto, U., & Nugraha, E. (2018). Comparative performance between C4. 5 and Naive Bayes classifiers in predicting student academic performance in a Virtual Learning Environment. In *2018 4th International Conference on Education and Technology (ICET)* (pp. 18–22). IEEE.

Bouguettaya, A., Yu, Q., Liu, X., Zhou, X., & Song, A. (2015). Efficient agglomerative hierarchical clustering. *Expert Syst. Appl.*, *42*(5), 2785–2797.

Dutt, A., Aghabozrgi, S., Ismail, M. A. B., & Mahroeian, H. (2015). Clustering algorithms applied in educational data mining. *Int. J. Inf. Electron. Eng.*, *5*(2), 112.

Gülagiz, F. K., & Sahin, S. (2017). Comparison of hierarchical and non-hierarchical clustering algorithms. *Int. J. Comput. Eng. Inf. Technol.*, *9*(1), 6.

Gupta, A., Sharma, H., & Akhtar, A. (n.d.). A COMPARATIVE ANALYSIS OF K-MEANS AND HIERARCHICAL CLUSTERING.

Hair Jr, J. F. (n.d.). Multivariate Data Analysis Joseph F. Hair Jr. William C. Black Barry J. Babin Rolph E. Anderson Seventh Edition.

Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. *J. R. Stat. Soc. Ser. c (Applied Stat.*, *28*(1), 100–108.

Hennig, C. (2007). Cluster-wise assessment of cluster stability. *Comput. Stat. Data Anal.*, *52*(1), 258–271.

Hennig, C. (2008). Dissolution point and isolation robustness: robustness criteria for general cluster analysis methods. *J. Multivar. Anal.*, *99*(6), 1154–1176.

Heuer, H., & Breiter, A. (2018). Student success prediction and the trade-off between big data and data minimization. *DeLFI 2018-Die 16. E-Learning Fachtagung Inform.*

Hlosta, M., Papathoma, T., & Herodotou, C. (2020). Explaining errors in predictions of at-risk students in distance learning education. In *International Conference on Artificial Intelligence in Education* (pp. 119–123). Springer.

Hussain, M., Zhu, W., Zhang, W., & Abidi, S. M. R. (2018). Student engagement predictions in an e-learning system and their impact on student course assessment scores. *Comput. Intell. Neurosci.*, *2018*.

Jaccard, P. (1901). Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines. *Bull Soc Vaudoise Sci Nat*, *37*, 241–272.

Karthikeyan, B., George, D. J., Manikandan, G., & Thomas, T. (2020). A comparative study on k-means clustering and agglomerative hierarchical clustering. *Int. J. Emerg. Trends Eng. Res.*, *8*(5).

Kaushik, M., & Mathur, B. (2014). Comparative study of K-means and hierarchical clustering techniques. *Int. J. Softw. Hardw. Res. Eng.*, *2*(6), 93–98.

Kuzilek, J., Hlosta, M., & Zdrahal, Z. (2017). Open university learning analytics dataset. *Sci. Data*, *4*(1), 1–8.

Likas, A., Vlassis, N., & Verbeek, J. J. (2003). The global k-means clustering algorithm. *Pattern Recognit.*, *36*(2), 451–461.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, pp. 281–297). Oakland, CA, USA.

McFadden, D. (1974). The measurement of urban travel demand. *J. Public Econ.*, *3*(4), 303–328.

McFadden, D. (1977). Modelling the choice of residential location.

Park, H.-S., & Jun, C.-H. (2009). A simple and fast algorithm for K-medoids clustering. *Expert Syst. Appl.*, *36*(2), 3336–3341.

Reddy, M., Makara, V., & Satish, R. (2017). Divisive Hierarchical Clustering with K-means and Agglomerative Hierarchical Clustering. *Int J Comp Sci. Trands Tech*, *5*(5), 5–11.

Rokach, L., & Maimon, O. (2005). Clustering methods. In *Data mining and knowledge discovery handbook* (pp. 321–352). Springer.

Rousseeuw, P. J., & Van Zomeren, B. C. (1990). Unmasking multivariate outliers and leverage points. *J. Am. Stat. Assoc.*, *85*(411), 633–639.

Templ, M., Filzmoser, P., & Reimann, C. (2008). Cluster analysis applied to regional geochemical data: problems and possibilities. *Appl. Geochemistry*, *23*(8), 2198–2213.

Verma, M., Srivastava, M., Chack, N., Diswar, A. K., & Gupta, N. (2012). A comparative study of various clustering algorithms in data mining. *Int. J. Eng. Res. Appl.*, *2*(3), 1379–1384.

Ward Jr, J. H. (1963). Hierarchical grouping to optimize an objective function. *J. Am. Stat. Assoc.*, *58*(301), 236–244.

Wook, M., Yahaya, Y. H., Wahab, N., Isa, M. R. M., Awang, N. F., & Seong, H. Y. (2009).

Predicting NDUM student's academic performance using data mining techniques. In *2009 Second International Conference on Computer and Electrical Engineering* (Vol. 2, pp. 357–361). IEEE.

Xu, D., & Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Ann. Data Sci.*, *2*(2), 165–193.

Yim, O., & Ramdeen, K. T. (2015). Hierarchical cluster analysis: comparison of three linkage measures and application to psychological data. *Quant. Methods Psychol.*, *11*(1), 8–21.

Table 1.
*List of all 18 process features*

| Feature Name | Description |
|---|---|
| dataplus | total clicks of a student on the additional informtion such as videos, audios, sites, etc. |
| dualpane | total clicks of a student on the information on site and activity related to that information |
| forumng | total clicks of a student on the discussion forum |
| glossary | total clicks of a student on the basic glossary related to contents of module |
| homepage | total clicks of a student on the module homepage |
| htmlactivity | total clicks of a student on the interactive html page |
| oucollaborate | total clicks of a student on the online video discussions |
| oucontent | total clicks of a student on the contents of the assignment |
| ouwiki | total clicks of a student on the Wikipedia content |
| page | total clicks of a student on the information related to module |
| questionnaire | total clicks of a student on the questionnaires related to module |
| quiz | total clicks of a student on the module quiz |
| resource | total clicks of a student on the pdf resources such as books |
| subpage | total clicks of a student on the other sites enabled in the module |
| url | total clicks of a student on the links to audio/video contents |
| duration | count of days of a student from the first day of having VLE activity to the last VLE activity day |
| active_days | count of days of a student having any VLE activity |
| first_grade | the student's first assignment's grade |

Table 2.

*Descriptive Statistics and Spearman Correlations among Process Features*

| Feature | *M* | *(SD)* | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. dataplus | 10.72 | (16.79) | -- | | | | | | | | | | | | | | | | |
| 2. dualpane | 0.78 | (1.46) | 0.64 | -- | | | | | | | | | | | | | | | |
| 3. forumng | 451.53 | (743.56) | 0.53 | 0.42 | -- | | | | | | | | | | | | | | |
| 4. glossary | 1.18 | (9.31) | 0.16 | 0.12 | 0.19 | -- | | | | | | | | | | | | | |
| 5. homepage | 465.58 | (468.12) | 0.63 | 0.51 | 0.82 | 0.16 | -- | | | | | | | | | | | | |
| 6. htmlactivity | 5.04 | (3.32) | 0.37 | 0.34 | 0.48 | 0.19 | 0.51 | -- | | | | | | | | | | | |
| 7. oucollaborate | 7.73 | (13.22) | 0.33 | 0.27 | 0.53 | 0.12 | 0.50 | 0.32 | -- | | | | | | | | | | |
| 8. oucontent | 1320.53 | (1033.6) | 0.69 | 0.57 | 0.64 | 0.14 | 0.82 | 0.47 | 0.40 | -- | | | | | | | | | |
| 9. ouwiki | 29.12 | (36.88) | 0.64 | 0.50 | 0.66 | 0.17 | 0.70 | 0.49 | 0.41 | 0.73 | -- | | | | | | | | |
| 10. page | 5.42 | (5.84) | 0.69 | 0.62 | 0.59 | 0.12 | 0.69 | 0.49 | 0.38 | 0.73 | 0.67 | -- | | | | | | | |
| 11. questionnaire | 11.75 | (13.98) | 0.77 | 0.64 | 0.59 | 0.11 | 0.68 | 0.44 | 0.34 | 0.74 | 0.74 | 0.73 | -- | | | | | | |
| 12. quiz | 762.72 | (577.83) | 0.37 | 0.28 | 0.52 | 0.12 | 0.64 | 0.36 | 0.40 | 0.56 | 0.40 | 0.42 | 0.41 | -- | | | | | |
| 13. resource | 39.69 | (28.45) | 0.45 | 0.37 | 0.58 | 0.20 | 0.66 | 0.45 | 0.45 | 0.60 | 0.54 | 0.51 | 0.47 | 0.54 | -- | | | | |
| 14. subpage | 298.15 | (197.62) | 0.61 | 0.51 | 0.72 | 0.19 | 0.89 | 0.53 | 0.46 | 0.86 | 0.67 | 0.69 | 0.65 | 0.68 | 0.70 | -- | | | |
| 15. url | 31.26 | (30.96) | 0.66 | 0.57 | 0.70 | 0.20 | 0.82 | 0.55 | 0.46 | 0.78 | 0.72 | 0.74 | 0.72 | 0.55 | 0.66 | 0.83 | -- | | |
| 16. duration | 247.56 | (61.97) | 0.42 | 0.31 | 0.56 | 0.12 | 0.64 | 0.33 | 0.37 | 0.52 | 0.41 | 0.43 | 0.43 | 0.51 | 0.47 | 0.56 | 0.54 | -- | |
| 17. active_days | 106.50 | (60.24) | 0.67 | 0.53 | 0.77 | 0.15 | 0.91 | 0.48 | 0.48 | 0.82 | 0.68 | 0.71 | 0.71 | 0.61 | 0.63 | 0.83 | 0.80 | 0.80 | -- |
| 18. first_grade | 79.17 | (17.02) | 0.20 | 0.17 | 0.32 | 0.04 | 0.29 | 0.15 | 0.17 | 0.25 | 0.28 | 0.21 | 0.27 | 0.24 | 0.18 | 0.25 | 0.26 | 0.27 | 0.28 |

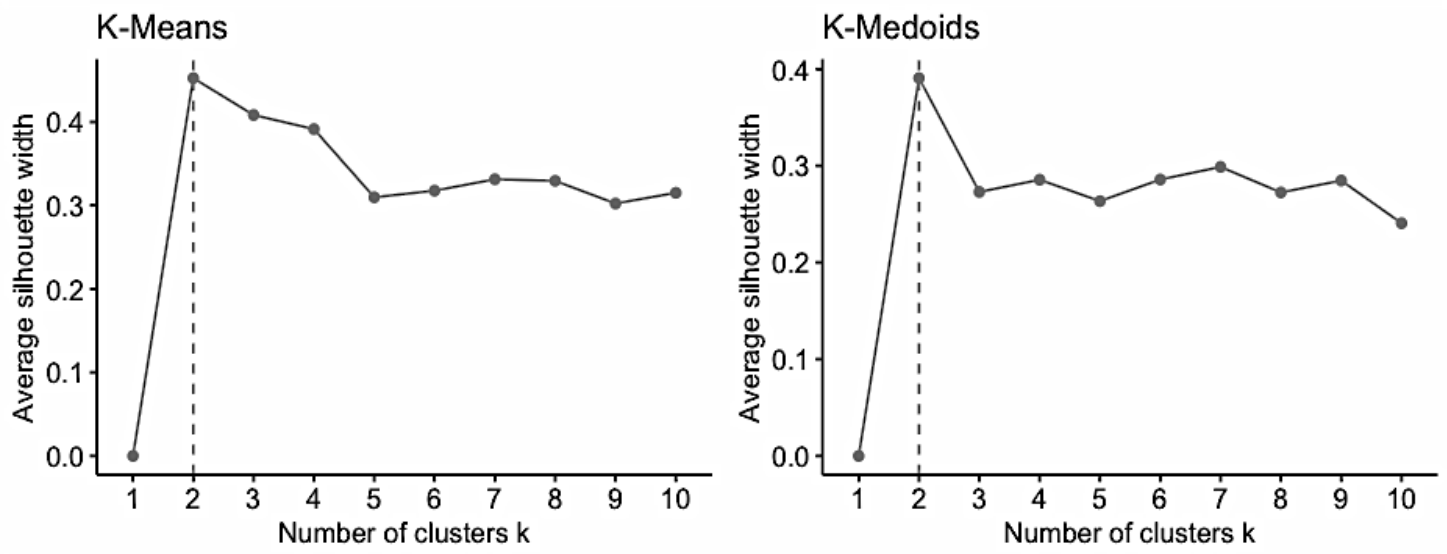


*Figure 1*. Average silhouette width values over different number of clusters (k) under k-means and k-medoids algorithms

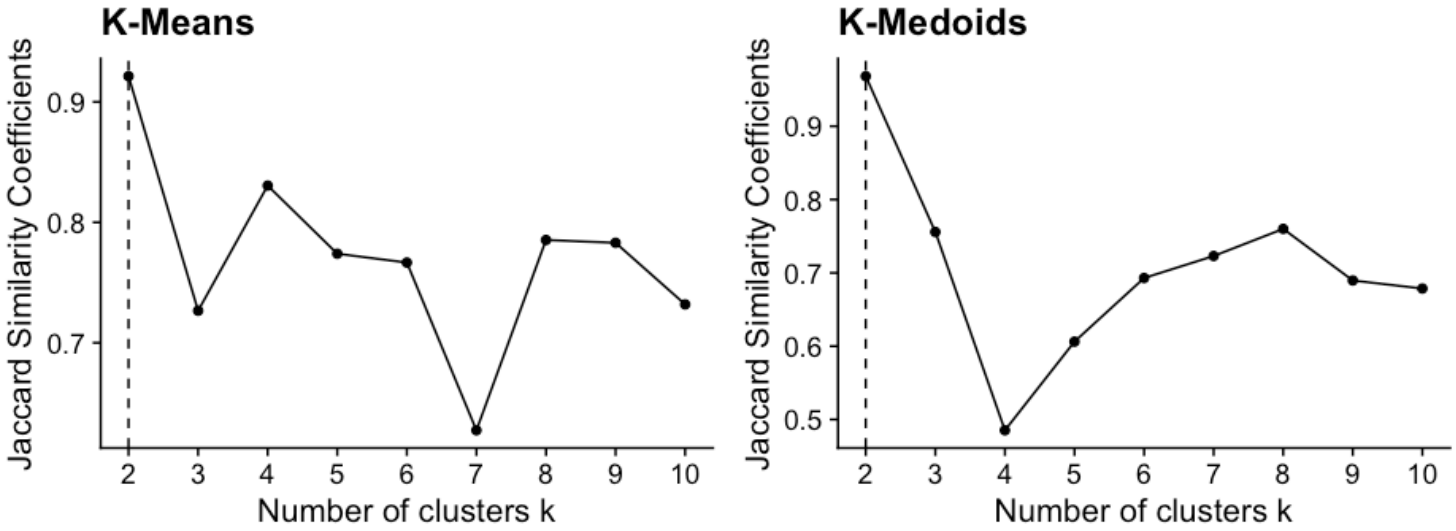


*Figure 2.* Average Jaccard similarity coefficients over different number of clusters (k) under k-means and k-medoids algorithms.

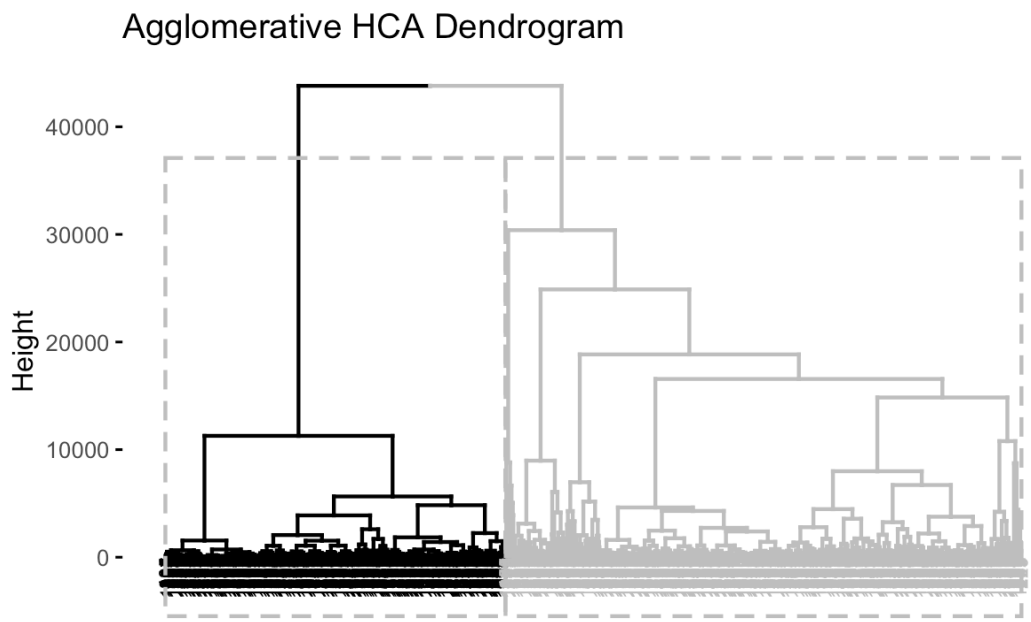


*Figure 3.* Dendrogram showing agglomerative HCA clustering results. Dendrogram were cut when two major clusters, represented by two different colors, were formed.

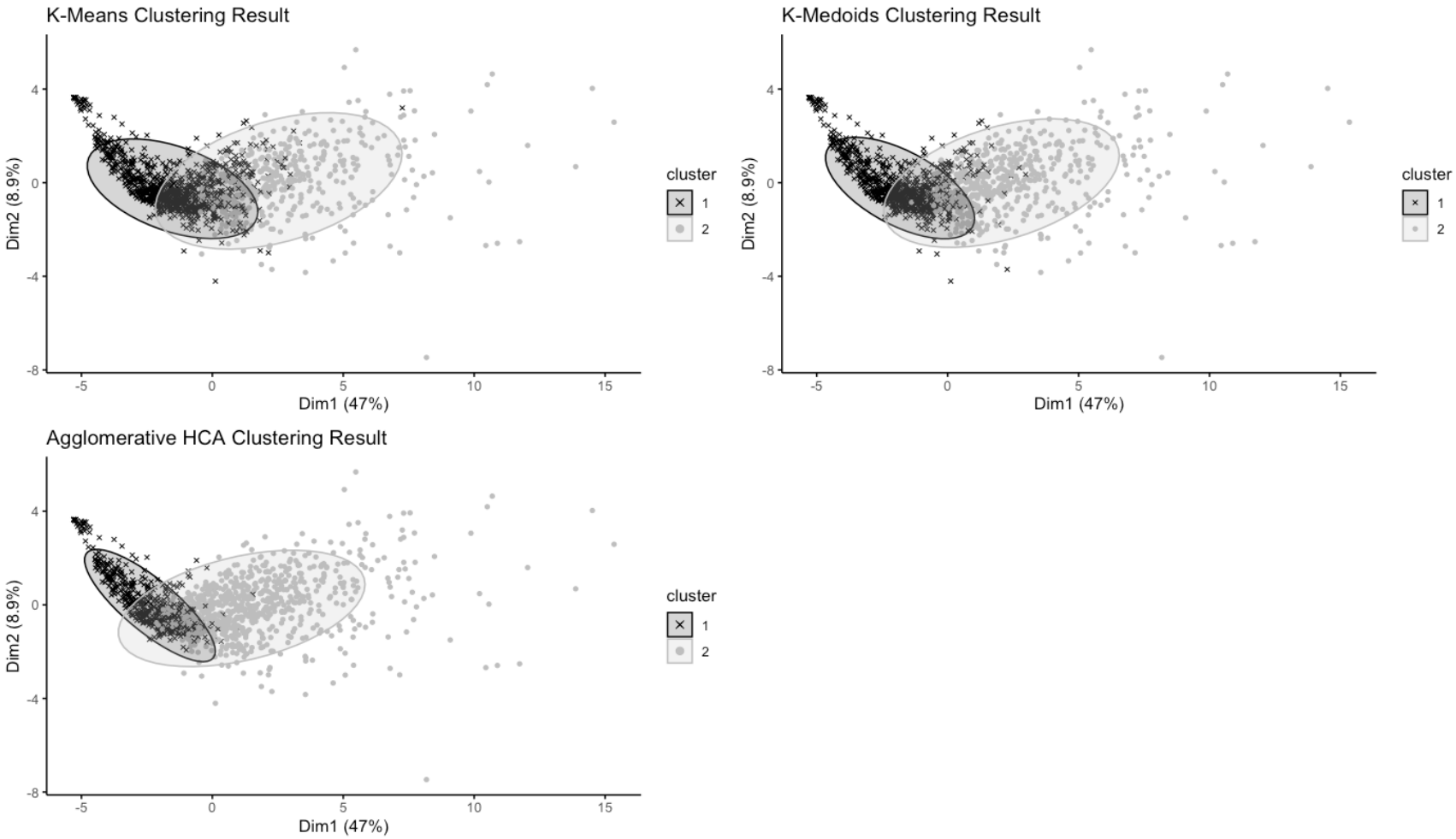


*Figure 4.* Clustering results for each algorithm. Each points represent an observation. Ellipse was drawn around each cluster.

Table 3.

*Comparison of k-means, k-medoids, and agglomerative HCA clustering results.*

| | *k* -Means | | | *k*-Medoids | | | Agglomerative HCA | | |
|---|---|---|---|---|---|---|---|---|---|
| | *Silhouette* | Clust1 | Clust2 | *Silhouette* | Clust1 | Clust2 | *Silhouette* | Clust1 | Clust2 |
| *Cluster Separation Indice* | 0.38 | | | 0.38 | | | 0.40 | | |
| Number of Students | | 871 | 428 | | 759 | 540 | | 516 | 783 |
| Percentage of Passing | | 0.69 | 0.96 | | 0.65 | 0.96 | | 0.51 | 0.95 |

*Note.* Sihouette represented the average silhouette width value of all the observations

Table 4.

*Logistic Regression Model Results for Predicting Module Outcome*

| | K-Means | | | | | K-Medoids | | | | | Agglomerative HCA | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\chi^2(1)$ | $R^2_{pseudo}$ | *BIC* | *b (SE)* | *OR* | $\chi^2(1)$ | $R^2_{pseudo}$ | *BIC* | *b (SE)* | *OR* | $\chi^2(1)$ | $R^2_{pseudo}$ | *BIC* | *b (SE)* | *OR* |
| *Model Fit* | 152.38 *** | 0.11 | 1238.97 | | | 208.61 *** | 0.15 | 1182.74 | | | 363.88 *** | 0.26 | 1027.48 | | |
| *Coefficients* | | | | | | | | | | | | | | | |
| Intercept | | | | 0.79 (0.07) *** | 2.20 | | | | 0.61 (0.08) *** | 1.84 | | | | 0.05 (0.09) | 1.05 |
| ClustMember | | | | 2.40 (0.26) *** | 10.98 | | | | 2.55 (0.23) *** | 12.78 | | | | 2.96 (0.19) *** | 19.25 |

*Note.* $N$=1299. Dependent variable, module outcome, was coded 1 for pass/distinction and 0 for fail. ClustMember = clustering groups assigned to students by each algorithm, which were dummy coded (i.e., cluster one as 0, cluster two as 1).

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$.

Table 5.

*Multiple Logistic Regression Model Results for Predicting Module Outcome*

| | Model 1 | | | Model 2 | | | Model 3 | | |
|---|---|---|---|---|---|---|---|---|---|
| | $\chi^2(1)$ | $R^2_{pseudo}$ | *b (SE)* | $\chi^2(1)$ | $R^2_{pseudo}$ | *b (SE)* | $\chi^2(1)$ | $R^2_{pseudo}$ | *b (SE)* |
| *Model Fit* | 98.04 *** | 0.06 | | 1629.60 *** | 0.93 | | 1648.50 *** | 0.94 | |
| *Coefficients* | | | | | | | | | |
| Intercept | | | 0.93 (0.31) ** | | | 13.77 (2.08) *** | | | 5.89 (561.92) |
| gender1 | | | -0.33 (0.08) *** | | | | | | 0.11 (0.38) |
| region1 | | | -0.27 (0.21) | | | | | | 0.30 (1.29) |
| region2 | | | -0.09 (0.18) | | | | | | 0.26 (0.82) |
| region3 | | | -0.14 (0.20) | | | | | | -4.19 (1.71) * |
| region4 | | | -0.27 (0.21) | | | | | | 0.60 (0.86) |
| region5 | | | 0.05 (0.21) | | | | | | 1.56 (1.05) |
| region6 | | | 0.13 (0.20) | | | | | | -0.86 (1.12) |
| region7 | | | -0.36 (0.20) | | | | | | 0.67 (1.19) |
| region8 | | | 0.16 (0.19) | | | | | | -0.67 (0.97) |
| region9 | | | 0.29 (0.21) | | | | | | 0.11 (1.19) |
| region10 | | | -0.03 (0.20) | | | | | | 2.37 (0.99) * |
| region11 | | | 0.22 (0.26) | | | | | | 0.18 (1.22) |
| region12 | | | 0.28 (0.18) | | | | | | -0.53 (0.88) |
| age_band1 | | | -0.65 (0.29) * | | | | | | 11.16 (561.92) |
| age_band2 | | | 0.45 (0.30) | | | | | | 12.39 (561.93) |
| disability1 | | | 0.11 (0.10) | | | | | | 0.30 (0.30) |
| dataplus | | | | | | 2.41 (1.13) * | | | 3.44 (1.40) * |
| dualpane | | | | | | -1.31 (0.52) * | | | -1.94 (0.67) ** |
| forumng | | | | | | 6.61 (1.56) *** | | | 8.75 (2.14) *** |
| glossary | | | | | | -0.01 (1.60) | | | -0.38 (1.41) |
| homepage | | | | | | 2.13 (1.72) | | | 1.95 (1.98) |
| htmlactivity | | | | | | -0.23 (0.38) | | | -0.35 (0.46) |
| oucollaborate | | | | | | 0.23 (0.65) | | | 0.02 (0.74) |
| oucontent | | | | | | 25.36 (3.65) *** | | | 33.19 (5.65) *** |
| ouwiki | | | | | | 0.04 (0.53) | | | 0.41 (0.63) |
| page | | | | | | 1.39 (0.77) | | | 2.20 (0.94) * |
| questionnaire | | | | | | 0.09 (0.89) | | | -0.40 (1.05) |
| quiz | | | | | | 4.33 (0.83) *** | | | 5.51 (1.11) *** |
| resource | | | | | | 0.40 (0.27) | | | 0.29 (0.33) |
| subpage | | | | | | -1.92 (0.92) * | | | -1.99 (1.05) |
| url | | | | | | 0.07 (0.66) | | | -0.49 (0.83) |
| active_days | | | | | | -1.19 (0.76) | | | -1.15 (0.92) |
| duration | | | | | | 0.08 (0.61) | | | 0.59 (0.82) |
| first_grade | | | | | | -0.01 (0.46) | | | 0.13 (0.59) |

*Note.* $N$=1299. Dependent variable, the cluster membership obtained from agglomerative HCA, was dummy coded. All the categorical variables (i.e., gender, region, age band, disability condition) were effect coded, and all 18 process feature variables were standardized to have a mean of 0 and variance of 1.

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$.

## Appendix (R Code)

```
#libraries
library(tidyverse) # data manipulation
library(psych) # model functions
library(cluster) # clustering algorithms
library(factoextra) # clustering algorithms & visualization
library(stats) # computing clustering info
library(fpc) # clusterboot

################################
### PART I: Module: FFF 2014J ###
################################

### Preprocessing Module Data ###
studentInfo <- studentInfo[(studentInfo$code_module %in% c("FFF")) &
(studentInfo$code_presentation == "2014J"),]
studentInfo <- studentInfo %>% filter((num_of_prev_attempts == 0) &
(final_result != 'Withdrawn'))
id <- studentInfo %>% select(id_student)

vle <- read.csv("data/openU/vle.csv")[,-c(5,6)]
vle <- vle[(vle$code_module %in% c("FFF")) & (vle$code_presentation ==
"2014J"),]
studentVle <- read.csv("data/openU/studentVle.csv")
studentVle <- studentVle[(studentVle$code_module %in% c("FFF")) &
(studentVle$code_presentation == "2014J"),]
studentVle <- studentVle %>%
  merge(., vle, by=c("id_site","code_module","code_presentation")) %>%
  merge(., id, by="id_student") # 1299

assess <- read.csv("data/OpenU/assessments.csv")
assess <- assess[(assess$code_module %in% c("FFF")) &
(assess$code_presentation == "2014J"),]
assess_stud <- read.csv("data/OpenU/studentAssessment.csv")
assess_stud <- assess_stud %>% merge(., assess, by="id_assessment")
assess_stud <- assess_stud[(assess_stud$code_module %in% c("FFF")) &
(assess_stud$code_presentation == "2014J"),]
assess_stud <- assess_stud %>% merge(., id, by="id_student")

### clicks variables ###
# across the whole module, how many times did the student click in this site
clicks <- studentVle %>%
  group_by(id_student, activity_type) %>%
  summarize(clicks = sum(sum_click))
clicks <- clicks %>% spread(activity_type,clicks) %>% replace(is.na(.),0)
```

```
describe(clicks[,-1])
multi.hist(clicks[,-1],density=FALSE, breaks=50, global=FALSE)

### first grade variable ###
first_grade <- assess_stud %>%
  group_by(id_student) %>%
  filter(date_submitted == min(date_submitted)) %>%
  summarize(first_grade = mean(score))

### active and duration variables ###
active <- studentVle %>% group_by(id_student, date) %>%
  summarize(n=n()) %>%
  group_by(id_student) %>%
  summarize(active_days=n())

duration <- studentVle %>% group_by(id_student) %>%
  summarize(last_day = max(date),
            first_day = min(date),
            duration = last_day - first_day + 1) %>%
  select(id_student, duration)

# Process feature variables ready
colnames(clicks)[2:17] <- paste(colnames(clicks)[2:17], 'clicks', sep='_')
process_features <- clicks %>%
  merge(., active, by='id_student') %>%
  merge(., duration, by='id_student') %>%
  merge(., first_grade, by="id_student", all.x=TRUE) %>%
  replace(is.na(.),0)
write.csv(overall, file='data/allFeatures.csv')




############################
### PART II: Clustering ###
############################

all <- read.csv("data/allFeatures.csv")
all <- all[, names(all) != "X"]
final <- all[1:2]
df <- all[3:21] %>% select(-repeatactivity_clicks)

### Boostrap to determine number of clusters ###
```

```
# k_means
km.jaccard <- vector("numeric", length=10)

for (i in 2:10){
  km.boot <- clusterboot(df, B=50, bootmethod="boot",
                         clustermethod=pamkCBI,
                         krange=i, seed=15555)
  km.jaccard[i] = mean(km.boot$bootresult)
}
km.jaccard <- km.jaccard[2:10]
km.k <- seq(2, 10, 1)
km.sta <- data.frame(cbind(km.jaccard, km.k))

# pam
pam.jaccard <- vector("numeric", length=10)
for (i in 2:10){
  pam.boot <- clusterboot(df, B=50, bootmethod="boot",
                         clustermethod=kmeansCBI,
                         krange=i, seed=15555)
  pam.jaccard[i] = mean(pam.boot$bootresult)
}
pam.jaccard <- pam.jaccard[2:10]
pam.k <- seq(2, 10, 1)
pam.sta <- data.frame(cbind(pam.jaccard, pam.k))

### K-means ###
k = 2
set.seed(123)
km.res <- kmeans(df, k, nstart=25) # try 25 different starting assignments
final <- cbind(final, k_means=km.res$cluster)

### K-medoids ###
set.seed(123)
pam.res <- pam(df, k, metric="manhattan")
final <- cbind(final, pam=pam.res$clustering)

### HCA ###
dist.eucl <- dist(df, method = "euclidean")
set.seed(123)
res.hc <- hclust(d = dist.eucl, method = "ward.D2")
grp <- cutree(res.hc, k=k)
final <- cbind(final, hca = grp)

### Graphs ###
plt1 <- fviz_cluster(km.res, data=df,
```

```
             palette = c("black","gray"), #c("#2E9FDF", "#E7B800"),
#"#00AFBB", "#FC4E07"),
             ellipse.type = "t",
             repel = TRUE,
             geom = "point",
             pointsize = 1,
             ggtheme = theme_classic(),
             main = "K-Means Clustering Result") +
             scale_shape_manual(values = c(4,19))
plt2 <- fviz_cluster(pam.res, data=df,
                     palette = c("black", "gray"),
             #palette = c("#2E9FDF", "#E7B800", "#00AFBB", "#FC4E07"),
             ellipse.type = "t",
             repel = TRUE,
             geom = "point",
             pointsize = 1,
             ggtheme = theme_classic(),
             main = "K-Medoids Clustering Result") +
             scale_shape_manual(values = c(4,19))
plt3 <- fviz_cluster(list(data = df, cluster = grp),
                     palette = c("black", "gray"),
             #palette = c("#2E9FDF", "#E7B800", "#00AFBB", "#FC4E07"),
             ellipse.type = "t",
             repel = TRUE,
             geom = "point",
             pointsize = 1,
             ggtheme = theme_classic(),
             main = "Agglomerative HCA Clustering Result")+
             scale_shape_manual(values = c(4,19))


list.plt <- list()
list.plt[[1]] <- plt1
list.plt[[2]] <- plt2
list.plt[[3]] <- plt3

plot_grid(plotlist = list.plt, ncol = 2)

fviz_dend(res.hc, k=k,
          cex = 0.5,
          k_colors = c("black", 'gray'),
          #k_colors = c("#2E9FDF","#E7B800"),
          color_labels_by_k = TRUE,
          rect = TRUE,
          main = "Agglomerative HCA Dendrogram")
```

```
##########################################
### PART III: Comparison and Validation ###
##########################################


############################
### Logistic Regression ###
############################

final <- final %>%  mutate(
  k_means = as.factor(k_means),
  pam = as.factor(pam),
  hca = as.factor(hca),
  final_2= as.factor(final_2)
)
log_km <- glm(final_2 ~ k_means, family="binomial", data=final)
log_pam <- glm(final_2 ~ pam, family="binomial", data=final)
log_hca <- glm(final_2 ~ hca, family="binomial", data=final)
summary(log_km)
summary(log_pam)
summary(log_hca)

###################################
### Multiple Logistic Regression ###
###################################

# df: 18 process variables - z-score
df_z <- data.frame(scale(df, center=TRUE, scale=TRUE))
# demo: merge with all -> studentInfo
demo <- studentInfo %>%
  merge(., data.frame(all$id_student),
        by.x="id_student",
        by.y="all.id_student", all.y=TRUE) %>%
  select(-c(1:3, 6, 7, 9, 10, 12)) # get rid of module and final result info
# demo: factor all the variables and effect coding them using contrasts
for (i in c(1:length(demo))){
  demo[i] = factor(demo[[i]], levels=unique(demo[[i]]),
labels=unique(demo[[i]]))
  contrasts(demo[[i]])=contr.sum(length(unique(demo[[i]])))
  print(i)
}
# outcome: final$hca -> dummy coding
hca_grp <- final %>%
  mutate(hca_grp = as_factor(dplyr::recode(hca, `1` = 0, `2` = 1))) %>%
```

```
  select(hca_grp)
# combine it together
profile_vars <- cbind(hca_grp, df_z, demo)

# model 1: demo
profile.m1 <- glm(hca_grp ~ gender + region + age_band + disability,
data=profile_vars, family = "binomial")
summary(profile.m1)

# model 2: process
profile.m2 <- glm(hca_grp ~ dataplus_clicks + dualpane_clicks +
forumng_clicks + glossary_clicks + homepage_clicks + htmlactivity_clicks +
oucollaborate_clicks + oucontent_clicks + ouwiki_clicks + page_clicks +
questionnaire_clicks + quiz_clicks + resource_clicks + subpage_clicks +
url_clicks + active_days + duration + first_grade, data=profile_vars, family
= "binomial")
summary(profile.m2)

# model 3: demo + process
profile.m3 <- glm(hca_grp ~ ., data=profile_vars, family = "binomial")
summary(profile.m3)
```