

MEC - Monitoring Clusters' Transitions

152

Abstract. In this paper we address the problem of monitoring the evolution of clusters, which has become an important research issue in recent years. We adopt two main strategies for cluster characterization - representation by enumeration and representation by comprehension -, and propose the MEC (Monitoring the Evolution of Clusters) framework, which was developed along the lines of the *change mining* paradigm. MEC includes a taxonomy of various types of cluster transitions, a tracking mechanism that depends on cluster representation, and a transition detection algorithm. Our tracking mechanism can be subdivided in two methods, devised to monitor clusters' transitions: one based on graphs' transitions, and another based on clusters' overlap. These are the most relevant contributions of this work. To demonstrate the feasibility and applicability of MEC we present illustrative examples, using datasets from different knowledge areas, such as Economy and Education. Our results are encouraging and demonstrate the ability of MEC framework to provide an efficient diagnosis of clusters' transitions.

1 INTRODUCTION

The celerity at which the evolution takes place, typically characterized by breaks and shifts, has increased exponentially in the last decades. The rapid progress made in science and technology has contributed to the emergence of a volatile and fast pace evolving world, which demands new perspectives in knowledge discovery upon data, such as time-oriented perspectives. The paradigm of *change mining* arises as a consequence of this evolution and encompasses Data Mining mechanisms that monitor models and patterns over time, compare them, detect and describe changes, and quantify them on their interestingness [1]. The challenge of *change mining* lays not only in the adaptation of models to changes in data distribution but also in the understanding of changes themselves. In this paper we propose MEC (Monitoring the Evolution of Clusters) framework, which is built along these lines, to tackle the problem of monitoring clusters' transitions over time, through the identification of temporal relationships among them. We assume that clusters can be represented in two different ways - by enumeration and by comprehension - and we adopt two main strategies to monitor and classify changes experienced by clusters, to each possible representation. The proposed methods were devised to deal only with clusters characterized by numerical attributes, however it is our intention to extend them to categorical attributes in the near future. Thus, our framework encompasses a taxonomy of various types of cluster transitions, that may be external or internal, a tracking mechanism that depends on cluster representation, and a transition detection algorithm.

The monitoring of the dynamics of clusters' structures is very important in many real world applications, since it fosters the cre-

ation of sustainable knowledge about the underlying phenomenon and, consequently, the adoption of pro-active attitudes. Besides, it may correlate to some important or critical events in the real applications or unveil the emergence of new ones. For these reasons, this study can benefit several areas, such as Marketing, Fraud Detection, Economy and Health. For example, the study of the evolution of customer's segments allow the detection of shifts in preferences and consumer habits, which can sustain the forecast of trends and consequent redefinition of Marketing's strategies and policies. The domain knowledge acquired by these means can act as a powerful differentiating factor in the market and strongly contributes to the creation, or reinforcement, of the company's competitive advantages. Traditional Data Mining is not able to help companies achieve these goals since it relies on static data and does not take into account its evolving nature. Therefore, the study of the dynamics of clusters contributes to the achievement of a greater understanding of cluster's evolutionary processes, broadening horizons and opening new paths in the way of thinking problems.

This paper is organized as follows. In Section 2 we provide a brief overview of the current state of the art. In Section 3 we distinguish two types of cluster representation and highlight their advantages and drawbacks. In Section 4 we formally introduce our MEC framework. In this section we present our taxonomy for clusters' transitions and explain the foundations of our tracking mechanism, for each type of representation. In Section 5 we show and discuss several experiments. Section 6 concludes our study.

2 RELATED WORK

Despite the extensive study of the clustering problem [3, 2], there is not much work conducted in the monitoring of cluster transitions. In this context of change, the research endeavor has been mainly directed to the adaptation of clusters to changed populations. However, the dynamic nature of most datasets encouraged new directions in research. This effort is clearly present in the areas of Machine Learning and Data Analysis.

There are several algorithms that directly or indirectly aim to capture and, especially, understand the dynamic nature of these datasets, particularly susceptible to the occurrence of changes in the underlying structure. The last decade has been especially profuse in the design of transition detection algorithms. Based on recent literature, it was possible to deduce a preliminary classification for algorithms built in this context. In general, there are algorithms designed to operate in relatively static (snapshots) or highly dynamic (Data Streams) environments. The first may be focused on the transitions experienced by generic patterns [4, 5, 7], clusters [8, 9, 10] or association rules [11, 12]. In the context of Data Streams, approaches focusing unclassified data are quite common (here we

include clustering algorithms for mobile objects or spatio-temporal objects). These approaches [14, 15, 16, 17, 18, 19, 20] elect clusters as its main data structure and are concerned with the efficiency and scalability of algorithms.

Based on our knowledge, currently there are two research works that address a problem related to ours [6, 8]. The framework proposed in [6] models the evolution of clusters by a weighted bipartite graph. MONIC framework [8] also proposes a cluster transition model to track cluster changes, supporting cluster comparisons across the time axis. MONIC uses a data ageing function that assigns lower weights to older observations and computes the overlap of clusters to capture its evolution. However, both frameworks only apply to clusters represented by enumeration, contrary to MEC, which is not restricted by the adopted method to represent clusters. MEC proposes a more complete framework to study the evolution of clusters which applicability to real world problems is much wider, since it can also consider clusters represented using summary statistics. This is the major contribution of our work, compared to the referred ones. Somewhat related is the work on novelty detection that can identify new concepts (in our terminology, the birth of new clusters), from unlabeled data (see for example [24]). These kind of methods are restricted to cluster's births, contrary to our framework, which is able to capture a much wider variety of concepts, such as splits, merges, deaths and survival.

3 CLUSTER CHARACTERIZATION

MEC framework assumes that clusters can be represented using two main strategies or representation schemes: **representation by enumeration** and **representation by comprehension** (can also be called extensional and intensional representation, using the mathematical nomenclature). In **representation by enumeration**, a cluster is characterized by its elements, ie, by the observations that were assigned to it by a given clustering algorithm (see Definition 1).

Definition 1 - REPRESENTATION BY ENUMERATION:

Let $\vec{x}_i$, ($i = 1, \dots, N$) be the i th observation defined as a vector of real numbers in a d -dimensional space $\vec{x}_i = (x_{i,1}, x_{i,2}, x_{i,3}, \dots, x_{i,d})$. A possible temporal representation of a cluster is defined as follows:

$$C_j(t) = \{\vec{x}_1, \dots, \vec{x}_m\}$$

where m is the number of observations assigned to cluster $C_j(t)$, $j = (1, \dots, k)$, k is the number of clusters and $t = (1, \dots, T)$, where T corresponds to the last analyzed timestamp.

This type of representation does not involve information loss and enables the monitoring of each observation over time, which contributes to the achievement of more reliable and accurate transition results. Though, it is not always possible to define clusters using this representation method, e.g. due to storage demands or privacy issues.

Alternatively, a cluster can be characterized through summary data, ie, using statistics that summarize its internal characteristics [24]. This is the idea behind **representation by comprehension** (see Definition 2), which is inspired by the notion of conceptual clustering.

Definition 2 - REPRESENTATION BY COMPREHENSION:

According to this representation a cluster C_j is a temporal object characterized by the following statistics:

$$C_j = \{ID, t, m, sup, r, \rho, \vec{c}\}$$

where ID is the unique identifier of C_j ($j = 1, \dots, k$), t is the time point where the cluster first appears ($t = 1, \dots, T$), m is the number of observations assigned to C_j , sup is the cluster's support, used to assess the relative importance of the cluster, r is the cluster's radius, ρ represents the density of cluster and $\vec{c}$ represents cluster's centroid.

To summarize the contents of each isolated cluster, within the comprehension representation scheme, we considered a centrality measure, a dispersion measure and a density measure. Due to simplicity issues, we adopted the cluster's centroid as our measure of centrality, the cluster's radius as our measure of dispersion and we use a definition of density that assumes spherical d -dimensional objects [24, 25]. Other measures could have been used, e.g. Mahalanobis metrics, but this would lead to more complex computations. The drawback of these measures is that they can only describe spherical or ball-shaped clusters¹. This kind of compact representations are very appealing to several real world applications, especially to those whose information are restricted due to privacy issues. However, it implies information loss, which may compromise the accuracy of *mapping* process². To implement the transitions detection algorithm we have considered the definitions of **radius** and **centroid** presented in [21]. Thereby, in this work, the **radius** of a cluster is the average distance from member points to the centroid.

4 MEC FRAMEWORK

We developed MEC framework in order to monitor the evolution of clusters' structures (also referred to as *clustering*) obtained at different time points. In this context, the concept of evolution refers to transitions undergone by clusters during time interval under observation $[t_i, t_{i+\delta}]$ ($\delta = 1, \dots, T - i$). Since there are, at least, two strategies to represent clusters, we designed a flexible tracking mechanism that is able to efficiently detect transitions experienced by clusters, regardless of how they are represented. Therefore, our tracking mechanism can be subdivided in two different methods, each one adapted to a distinct cluster representation. In the next subsections we present our taxonomy of transitions and the two developed methods.

4.1 Taxonomy of Transitions

There are at least eight taxonomic schemes for the classification of transitions in clusters, patterns or concepts that evolve over time, in the literature [9, 14, 16, 15, 22, 20, 8, 10]. To capture the changes likely to occur in clusters' structures we considered the following taxonomy: **Birth**, **Death**, **Split**, **Merge** and **Survival** of clusters. These transitions are external, as they relate to changes in the whole *clustering*. The key concept for the detection and evaluation of these transitions is the concept of *mapping*, which can be defined as the process of discovering the matches between clusters obtained at time

¹ In this work, we restrict to numerical features. For non-numerical features we will need appropriate measures for centrality and dispersion.

² *Mapping* is the process of discovering the matches of clusters between two distinct time points

point t_i and clusters obtained at a later time point $t_{i+\delta}$, in case they still exist. On the other hand, it is also possible to categorize internal transitions, ie, changes concerning the contents of each isolated cluster. For this purpose, we considered size and compactness transitions [8]. Size transitions encompasses **Expansion** or **Shrinkage**, depending on the increase or decrease of the surviving cluster's cardinality. Compactness transitions include **Compression** or **Dispersal**, which depends on the increase or decrease of the surviving cluster's density. These type of transitions can be easily traced through the monitoring of summary data, such as the cardinality and the density of the surviving clusters.

4.2 Method for Clusters Represented by Enumeration

In this method, the *mapping* process explores the concept of conditional probability and is restricted by a pre-defined threshold - **survival threshold** τ -, which assumes the minimum of $\tau = 0.5$. The use of conditional probabilities requires structurally identical datasets, ie, datasets composed by observations of the same objects, in each time point under analysis. These conditional probabilities are computed for every pair of possible connections between clusters obtained at different time points and they represent the edge's weights in a bipartite graph. The use of bipartite graphs is related to its usefulness in modeling matching problems and the fact that graph based representations are visually appealing, exploiting the power of the eye and human intuition. The foundations of our transitions detection algorithm are based in this idea, which can be defined as follows (Definition 3).

Definition 3 - WEIGHTED BIPARTITE GRAPHS:

Given the clusterings $\xi_i, \xi_{i+\delta}$, obtained at $t_i, t_{i+\delta}$, a graph $G = (U, V, E)$ can be constructed, where U represents the first subset of vertices (clusters of t_i), V represents the second subset of vertices (clusters of $t_{i+\delta}$), and E denotes a set of weighted edges between any pair of clusters belonging to ξ_i and $\xi_{i+\delta}$. Formally, the weight assigned to the edge connecting clusters $C_m(t_i)$ and $C_u(t_{i+\delta})$ ($m = (1, \dots, k_{t_i})$ and $u = (1, \dots, k_{t_{i+\delta}})$, where k_{t_i} and $k_{t_{i+\delta}}$ are the number of clusters returned by a given clustering algorithm in timepoints t_i and $t_{i+\delta}$, respectively) are estimated in accordance with the conditional probability (Equation 1):

$$\begin{aligned} \text{weight}(C_m(t_i), C_u(t_{i+\delta})) &= P(X \in C_u(t_{i+\delta}) | X \in C_m(t_i)) = \\ &= \frac{\sum P(x \in C_m(t_i) \cap C_u(t_{i+\delta}))}{\sum P(x \in C_m(t_i))} \end{aligned}$$

where X is the set of observations assigned to cluster $C_m(t_i)$ and $P(X \in C_u(t_{i+\delta}) | X \in C_m(t_i))$ represents the probability of X belonging to cluster C_u from $t_{i+\delta}$ knowing that X belongs to cluster C_m obtained at a previous timestamp t_i .

To detect changes, we formally define the transitions that a cluster $C \in \xi_i$ can experience, with respect to $\xi_{i+\delta}$. It was introduced a new threshold to help the definition of these transitions: the **split threshold** λ . This formal design is based on MONIC's framework's external transitions [8] and is depicted in Table 1. The drawback of this method boils down to the fact that monitoring based on graph transitions only allow the detection of external transitions. However, internal transitions can be detected in this method through the computation of statistics, like density and cardinality of clusters.

4.3 Method for Clusters Represented by Comprehension

For compact representations of clusters, we developed a method to assess the similarity between clusters through the discovery of an intersection region in the feature space, formed by pairs of clusters from different timestamps. This procedure allows us to know if cluster $C_m(t_i)$ in clustering ξ_i is the match of cluster $C_u(t_{i+\delta})$ in the later clustering $\xi_{i+\delta}$. To accomplish this task, we compute the Euclidean distance between clusters' centroids (see Definition 4) for every pair of clusters belonging to clusterings generated at different time points.

Definition 4 - CENTROID EUCLIDEAN DISTANCE:

Let C_m and C_u be two clusters obtained at t_i and $t_{i+\delta}$, respectively, and let $\vec{c}_m = (c_{m,1}, c_{m,2}, \dots, c_{m,d})$ and $\vec{c}_u = (c_{u,1}, c_{u,2}, \dots, c_{u,d})$ be the corresponding centroids. The Centroid Euclidean Distance is used to measure the similarity between $C_m(t_i)$ and $C_u(t_{i+\delta})$, and is defined as:

$$d(\vec{c}_m, \vec{c}_u) = \sqrt{\sum_{j=1}^d (c_{m,j} - c_{u,j})^2} \quad (1)$$

Posteriorly, to assess the existence of overlap (or intersection) between clusters, we compare the centroids' distance with the sum of the radius of clusters. If the centroids' distance is equal or greater than the sum of clusters' radius (2), then we can deduce that clusters don't intersect and, therefore, the cluster of t_i cannot be the match of the cluster of $t_{i+\delta}$. Otherwise (3), we can assume that these pair of clusters intersect in the feature space and conclude that they form a match.

$$d(C_m(t_i), C_u(t_{i+\delta})) \geq r_{C_m(t_i)} + r_{C_u(t_{i+\delta})} \quad (2)$$

$$d(C_m(t_i), C_u(t_{i+\delta})) < r_{C_m(t_i)} + r_{C_u(t_{i+\delta})} \quad (3)$$

To detect external and internal transitions and discover the changes undergone by clusters, we formally define the transitions, using the presented concepts. The formal design is depicted in Table 2 and Table 3. Note that internal transitions are only monitored for surviving clusters and, also, that the usage of Euclidean distance recommends the standardization of the attributes.

The implementation of the transitions detection algorithm, in software R 2.10.0, was supported by the definitions presented in Table 1, Table 2 and Table 3.

To better understand this framework, in next section we present real world case studies.

5 REAL WORLD CASE STUDIES

In order to show the feasibility and application of the proposed framework and glean insights about the evolution of clusters, we experimented MEC in real datasets extracted from Banco de Portugal's Central Balance-Sheet Database (CBSD) and from the Portuguese Institute of Statistics (INE).

Table 1. Formal definition of the external transitions of a cluster represented by enumeration

Transitions' Taxonomy	Notation	Formal Definition
Cluster's Birth	$\emptyset \rightarrow C_u(t_{i+\delta})$	$0 < weight(C_m(t_i), C_u(t_{i+\delta})) < \tau \forall_m$
Cluster's Death	$C_m(t_i) \rightarrow \emptyset$	$weight(C_m(t_i), C_u(t_{i+\delta})) < \lambda \forall_u$
Cluster's Split	$C_m(t_i) \xrightarrow{\subseteq} \{C_1(t_{i+\delta}), \dots, C_r(t_{i+\delta})\}$	$(\exists_u \exists_v : weight(C_m(t_i), C_u(t_{i+\delta})) \geq \lambda \wedge weight(C_m(t_i), C_v(t_{i+\delta})) \geq \lambda) \wedge \sum_{u=1}^r weight(C_m(t_i), C_u(t_{i+\delta})) \geq \tau$
Cluster's Merge	$\{C_1(t_i), \dots, C_p(t_i)\} \xrightarrow{\subseteq} C_u(t_{i+\delta})$	$(weight(C_m(t_i), C_u(t_{i+\delta})) \geq \tau) \wedge \exists C_p \in \xi_i \setminus \{C_m\} : weight(C_p(t_i), C_u(t_{i+\delta})) \geq \tau$
Cluster's Survival	$C_m(t_i) \rightarrow C_u(t_{i+\delta})$	$(weight(C_m(t_i), C_u(t_{i+\delta})) \geq \tau) \wedge \nexists C_p \in \xi_i \setminus \{C_m\} : weight(C_p(t_i), C_u(t_{i+\delta})) \geq \tau$

Table 2. Formal definition of the external transitions of a cluster represented by comprehension

Transitions' Taxonomy	Notation	Formal Definition
Cluster's Birth	$\emptyset \rightarrow C_u(t_{i+\delta})$	$d(C_m(t_i), C_u(t_{i+\delta})) \geq r_{C_m(t_i)} + r_{C_u(t_{i+\delta})} \forall_m$
Cluster's Death	$C_m(t_i) \rightarrow \emptyset$	$d(C_m(t_i), C_u(t_{i+\delta})) \geq r_{C_m(t_i)} + r_{C_u(t_{i+\delta})} \forall_u$
Cluster's Split	$C_m(t_i) \xrightarrow{\subseteq} \{C_1(t_{i+\delta}), \dots, C_r(t_{i+\delta})\}$	$(d(C_m(t_i), C_u(t_{i+\delta})) < r_{C_m(t_i)} + r_{C_u(t_{i+\delta})}) \wedge \exists C_r \in \xi_{i+\delta} \setminus \{C_u\} : d(C_m(t_i), C_r(t_{i+\delta})) < r_{C_m(t_i)} + r_{C_r(t_{i+\delta})}$
Cluster's Merge	$\{C_1(t_i), \dots, C_p(t_i)\} \xrightarrow{\subseteq} C_u(t_{i+\delta})$	$(d(C_m(t_i), C_u(t_{i+\delta})) < r_{C_m(t_i)} + r_{C_u(t_{i+\delta})}) \wedge \exists C_p \in \xi_i \setminus \{C_m\} : d(C_p(t_i), C_u(t_{i+\delta})) < r_{C_p(t_i)} + r_{C_u(t_{i+\delta})}$
Cluster's Survival	$C_m(t_i) \rightarrow C_u(t_{i+\delta})$	$(d(C_m(t_i), C_u(t_{i+\delta})) < r_{C_m(t_i)} + r_{C_u(t_{i+\delta})}) \wedge \nexists C_p \in \xi_i \setminus \{C_m\} : d(C_p(t_i), C_u(t_{i+\delta})) < r_{C_p(t_i)} + r_{C_u(t_{i+\delta})}$

5.1 REPRESENTATION BY ENUMERATION

5.1.1 First Experiment - Portuguese Activity Sectors

For the first experiment, we extracted three datasets from CBSD. Each dataset corresponds to a year (2005, 2006 and 2007) and consists of 439 observations characterized by 10 continuous attributes. The objects represent activity sectors, according to the higher granularity level of CEA (Portuguese Classification of Economic Activities), and the attributes are financial and economic aggregated indicators (e.g. net income, investment rate and labor productivity). It should be noted that the activity sectors and performance indicators are exactly the same, for all periods under analysis, which is a requirement for the application of MEC framework. The attributes were normalized using Z-scores, due to significant differences on scale and dispersion. In order to discover the clusters (input of our study) from the CBSD' datasets, we conducted experiments using two different algorithms for clustering: the agglomerative hierarchical algorithm using Ward's method, and K-means partitional algorithm. Although there were experiments with other algorithms, the mentioned ones have achieved good quality levels. In both algorithms, the determination of the critical clustering structure, through the identification of the best number of clusters (best k), was supported by the analysis of an internal validation measure: the silhouette width [23]. Afterwards, we applied our transitions detection algorithm, setting a fixed $\lambda = 0.2$ and $\tau = 0.5$, whose choice was guided by a thresholds' sensitivity analysis. These parameters are more relaxed and allow the detection of a wider variety of transitions. The resulting graphs, showing the transitions between clusters, are depicted in Figure 1.

On the left-hand side of Figure 1 we observe two bipartite graphs, one corresponding to transitions experienced during the time interval [2005, 2006] and another one corresponding to transitions occurred during [2006, 2007]. In the former, there were two surviving clus-

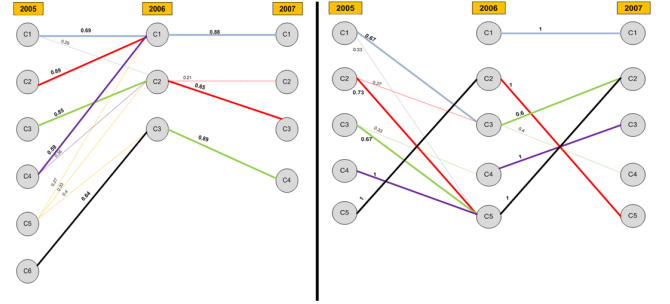


Figure 1. Bipartite graphs, corresponding to timestamps 2005, 2006 and 2007, with the thickness of the edges indicating weights that are equal or greater to survival threshold; the edges whose weights are below the split threshold were removed from the graphs, due to their insignificance. The left-hand side graph corresponds to the agglomerative hierarchical clustering and the right-hand side to the K-means clustering.

Table 3. Formal definition of the internal transitions of a cluster represented by comprehension

Transitions' Taxonomy	Notation	Formal Definition
Cluster's Expansion	$C_m(t_i) \nearrow C_u(t_{i+\delta})$	$\#C_m(t_i) < \#C_u(t_{i+\delta})$
Cluster's Shrinkage	$C_m(t_i) \searrow C_u(t_{i+\delta})$	$\#C_m(t_i) \geq \#C_u(t_{i+\delta})$
Cluster's Compression	$C_m(t_i) \xrightarrow{\bullet} C_u(t_{i+\delta})$	$\lambda_{C_m(t_i)} < \lambda_{C_u(t_{i+\delta})}$
Cluster's Dispersal	$C_m(t_i) \xrightarrow{*} C_u(t_{i+\delta})$	$\lambda_{C_m(t_i)} > \lambda_{C_u(t_{i+\delta})}$

ters ($C_3(2005) \rightarrow C_2(2006)$ and $C_6(2005) \rightarrow C_3(2006)$), a merge of three clusters ($\{C_1(2005), C_2(2005), C_4(2005)\} \searrow C_1(2006)$) and a split ($C_5(2005) \searrow \{C_1(2006), C_2(2006), C_3(2006)\}$). In the latter, all clusters survived ($C_1(2006) \rightarrow C_1(2007)$, $C_2(2006) \rightarrow C_3(2007)$ and $C_3(2006) \rightarrow C_4(2007)$) and a new one has emerged ($\emptyset \rightarrow C_2(2007)$). The right-hand side of Figure 1 corresponds to the K-means algorithm's bipartite graphs. Using the same strategy and the silhouette width validation measure, K-means algorithm resulted in a different partition (five clusters to all time points under analysis). The detected transitions also differ from the ones obtained for the agglomerative hierarchical algorithm, as expected. Thereby, during [2005, 2006] we detect the survival of two clusters ($C_1(2005) \rightarrow C_3(2006)$ and $C_5(2005) \rightarrow C_2(2006)$), a merge of three clusters ($\{C_2(2005), C_3(2005), C_4(2005)\} \searrow C_5(2006)$) and two births ($\emptyset \rightarrow C_1(2006)$ and $\emptyset \rightarrow C_4(2006)$). At period [2006, 2007], four clusters survived ($C_1(2006) \rightarrow C_1(2007)$, $C_2(2006) \rightarrow C_5(2007)$ and $C_4(2006) \rightarrow C_3(2007)$), two clusters merged ($\{C_3(2006), C_5(2006)\} \searrow C_2(2007)$) and a new one has emerged ($\emptyset \rightarrow C_4(2007)$).

In this experiment, the differences in terms of transitions between the hierarchical and the K-means algorithm are explained by the different number of clusters assumed by each algorithm. If k was the same for each algorithm, transitions results would converge. The merge occurred in [2005, 2006] was captured by both clustering algorithms, so we assumed this was an important transition. The inspection of the dataset suggested that these three clusters were grouped into one because the activity sectors assigned to them experienced a worsening in their economic and financial performance, which was reflected in the mitigation of their initial differences. But why did this happen? To answer this question we searched for relevant information about the topic and we found out that, in 2006, there has been a homogenization of the information process within the SIMPLEX portuguese program, which aimed to incorporate in one document (the Simplified Business Information) and a single delivery operation the information that companies are required to provide to public institutions. Moreover, the reporting of the Business information became mandatory, which contributed to the increase of the coverage degree of CBSD. For this reason, the data became more reliable and complete, reflecting a more realistic image of the country, which may be the cause of the detected cluster's merge.

5.1.2 Second Experiment - Students Enrolled in Non-Higher Education

The second experiment was designed using datasets extracted from the Portuguese Institute of Statistics (INE). These datasets focus on the number of students enrolled in non-higher education, which are available for years 2001, 2002 and 2003, and are useful for the analysis of a country's Education. Each dataset consists of 30 observations, corresponding to the units of analysis of NUTS III (Nomenclature of territorial units for statistics level III), and 5 continuous attributes

that are expressed in terms of the number of enrolled students in each non-higher school stage. Similarly to the first experiment, the observations and attributes are exactly the same for both years. Experimental conditions are also the same. The resulting graphs, that illustrate the transitions, are depicted in Figure 2.

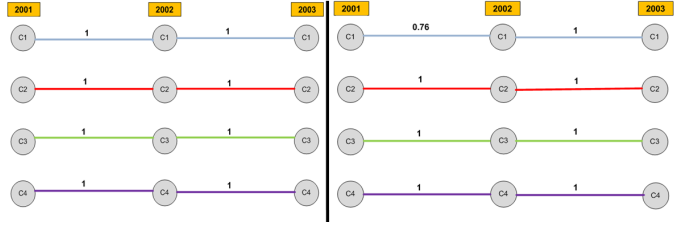


Figure 2. Bipartite graphs, corresponding to timestamps 2001, 2002 and 2003. The left-hand side graph corresponds to the agglomerative hierarchical clustering and the right-hand side to the K-means clustering.

The analysis of Figure 2 allows the diagnosis of changes between three snapshots of data (2001, 2002 and 2003). As evident from the bipartite graphs, all clusters survived. It is also evident that both clustering algorithms agree that there were no structural changes over the years. This clearly indicates that there were no significant changes in the number of enrolled students, in the time period under analysis. From this we can deduce that, in this time interval, Portugal didn't implement remarkable Education reforms.

5.2 REPRESENTATION BY COMPREHENSION

To test the tracking mechanism for compact representations of clusters we use the same dataset from CBSD. Since the data contain information on all observations, when applying the clustering algorithm we transformed the original representation into a comprehension representation scheme, according to Definition 2. We show the results using the same clustering algorithms - hierarchical algorithm and K-means -, and briefly compare the solutions obtained for each representation scheme (by enumeration and by comprehension).

During [2005, 2006], and using the hierarchical agglomerative clustering algorithm (partition in six, three and four clusters, for each year), the transitions detection algorithm detected the merge of three groups of clusters ($\{C_1(2005), C_2(2005)\} \searrow C_1(2006)$, $\{C_1(2005), C_3(2005)\} \searrow C_2(2006)$ and $\{C_5(2005), C_6(2005)\} \searrow C_3(2006)$), a split ($C_1(2005) \searrow \{C_1(2006), C_2(2006)\}$) and the death of one cluster ($C_4(2005) \rightarrow \emptyset$). At time interval [2006, 2007] we observe the split of two groups of clusters ($C_1(2006) \searrow \{C_1(2007), C_4(2007)\}$ and $C_2(2006) \searrow \{C_1(2007), C_2(2007), C_3(2007), C_4(2007)\}$) and two merges ($\{C_1(2006), C_2(2006)\} \searrow C_1(2007)$ and

$\{C_1(2006), C_2(2006), C_3(2006)\} \xrightarrow{S} C_4(2007)$). Using the K-means algorithm we obtain a partition in five clusters, and different transitions results. During [2005, 2006] a cluster was born ($\emptyset \rightarrow C_1(2006)$), four clusters died ($C_3(2005) \rightarrow \emptyset$, $C_4(2005) \rightarrow \emptyset$, $C_5(2005) \rightarrow \emptyset$ and $C_6(2005) \rightarrow \emptyset$) and two clusters survived ($C_1(2005) \rightarrow C_3(2006)$ and $C_2(2005) \rightarrow C_2(2006)$). For the surviving clusters we investigated the possible existence of internal transitions. The detected internal transitions were only related to changes in clusters' size. Thus, we found that cluster $C_1(2005)$ shrinks ($C_1(2005) \searrow C_3(2005)$), reducing the number of observations assigned to it, and that cluster $C_2(2005)$ suffers the reverse process - *expansion* - increasing its cardinality ($C_1(2005) \nearrow C_2(2006)$). During [2006, 2007], we detect the death of one cluster ($C_1(2006) \rightarrow \emptyset$), the birth of two clusters ($\emptyset \rightarrow C_2(2007)$ and $\emptyset \rightarrow C_4(2007)$) and the survival of another two clusters ($C_2(2006) \rightarrow C_3(2007)$ and $C_3(2006) \rightarrow C_1(2007)$). Both surviving clusters shrink ($C_2(2006) \searrow C_3(2007)$ and $C_3(2006) \searrow C_1(2007)$).

Comparing the results obtained for the two different representation schemes we can deduce that the transitions detection algorithm's behavior differs. The most obvious difference is the number of deaths and births, which is much higher in the method of representation by comprehension. These differences are related to the foundations of each method and the different strategies used for mapping the clusters.

6 CONCLUSION

In this paper we introduced a framework for addressing the problem of monitoring and detecting transitions in clusters' structures. We define and propose two different strategies to represent clusters. We also expose two methods, adapted to each type of cluster representation. Two real world case studies are presented to illustrate the applicability and feasibility of MEC. In these illustrative examples the framework proved its ability to provide an effective diagnosis of the transitions experienced by clusters.

The novelty of these methods lies in the use of bipartite graphs and conditional probabilities to monitor the transitions of clusters represented by enumeration, and in the computation and assessment of clusters' overlap, for compact representation of clusters.

As future work we intend to extend the application domain of MEC to categorical attributes, explore more complex intensional representation schemes and develop a similar framework to monitor the evolution of social networks.

Acknowledgements: Thanks to the support of the project Knowledge Discovery from Ubiquitous Data Streams (PTDC/EIA-EIA/098355/2008).

REFERENCES

- [1] Bottcher, M., Hoppner, F., Spiliopoulou, M. : On exploiting the power of time in data mining. SIGKDD Explorations (10), 3–11 (2008)
- [2] Jain, A. K. : Data Clustering: 50 Years Beyond K-means. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML/PKDD 2008. LNCS, vol. 5211. Springer, Belgium (2008)
- [3] Jain, A. K., Murty, M. N., Flynn, P. J. : Data Clustering: A Review. ACM Comput. Surv. (31), 264–323 (1999)
- [4] Ganti, V., Gehrke, J., Ramakrishnan, R. : A Framework for Measuring Changes in Data Characteristics. In: Proceedings of the 18th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp. 126–137. ACM Press, Pennsylvania (1999)
- [5] Bartolini, I., Ciaccia, P., Ntoutsi, I., Patella, M., Theodoridis, Y. : The Panda framework for Comparing Patterns. Data Knowl. Eng. (68), 244–260 (2009)
- [6] Oliveira, M., Gama, J. : Bipartite Graphs for Monitoring Clusters Transitions. In: Proceedings of the Ninth International Symposium on Intelligent Data Analysis. LNCS, vol. 6065, pp. to appear. Springer, Arizona (2010)
- [7] Chawathe, S. S., Garcia-Molina, H. : Meaningful Change Detection in Structured Data. In: Peckham, J. (ed.) Proceedings ACM SIGMOD International Conference on Management of Data, pp. 26–37. ACM Press, Arizona (1997)
- [8] Spiliopoulou, M., Ntoutsi, I., Theodoridis, Y., Schult, R. : MONIC: modeling and monitoring cluster transitions. In: Eliassi-Rad, T., Ungar, L. H., Craven, M., Gunopulos, D. (eds.) ACM SIGKDD 2006, pp. 706–711. ACM, Philadelphia (2006)
- [9] Falkowski, T., Bartelheimer, J., Spiliopoulou, M. : Mining and Visualizing the Evolution of Subgroups in Social Networks. In: IEEE / WIC / ACM International Conference on Web Intelligence, pp. 52–58. IEEE Computer Society, China (2006)
- [10] Yang, H., Parthasarathy, S., Mehta, S. : A generalized framework for mining spatio-temporal patterns in scientific data. In: Grossman, R., Baryardo, R. J., Bennett, K. P. (eds.) Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 716–721. ACM, Illinois (2005)
- [11] Baron, S., Spiliopoulou, M. : Monitoring Change in Mining Results. In: Kambayashi, Y., Winiwarter, W., Arikawa, M. (eds.) Data Warehousing and Knowledge Discovery, Third International Conference. LNCS, vol. 2114. Springer, Germany (2001)
- [12] Baron, S., Spiliopoulou, M. : Monitoring the Evolution of Web Usage Patterns. In: Berendt, B., Hotho, A., Mladenic, D., Someren, M., Spiliopoulou, M., Stumme, G. (eds.) Web Mining: From Web to Semantic Web, First European Web Mining Forum. LNCS, vol. 3209. Springer, Croatia (2004)
- [13] Lu, Y.-H., Huang, Y. : Mining data streams using clustering. In: Proceedings of the 4th International Conference on Machine Learning and Cybernetics, pp. 2079–2083. IEEE Computer Society, China (2005)
- [14] Aggarwal, C. C. : On Change Diagnosis in Evolving Data Streams. IEEE Trans. Knowl. Data Eng. (17), 587–600 (2005)
- [15] Chen, K., Liu, L. : Detecting the Change of Clustering Structure in Categorical Data Streams. In: Ghosh, J., Lambert, D., Skillicorn, D. B., Srivastava, J. (eds.) Proceedings of the 6th SIAM International Conference on Data Mining. SIAM, USA (2006)
- [16] Aggarwal, C. C., Han, J., Wang, J., Yu, P. S. : A Framework for Change Diagnosis of Data Streams. In: Halevy, A. Y., Ives, Z. G., Doan, A. : Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pp. 575–586. ACM, California (2003)
- [17] O'Callaghan, L., Meyerson, A., Motwani, R., Mishra, N., Guha, S. : Streaming-Data Algorithms for High-Quality Clustering. In: Proceedings of the 18th International Conference on Data Engineering. IEEE Computer Society, California (2002)
- [18] Elnekave, S., Last, M., Maimon, O. : Incremental Clustering of Mobile Objects. ICDE Workshops (2007)
- [19] Kalnis, P., Mamoulis, N., Bakiras, S. : On Discovering Moving Clusters in Spatio-temporal Data. In: Medeiros, C. B., Egenhofer, M. J., Bertino, E. (eds.) Advances in Spatial and Temporal Databases, 9th International Symposium. LNCS, vol. 3633, pp. 364–381. Springer, Brazil (2005)
- [20] Li, T., Ma, S., Ogiwara, M. : Entropy-based criterion in categorical clustering. In: Proceedings of the 21th international conference on Machine learning. ACM, New York (2004)
- [21] Zhang, T., Ramakrishnan, R., Livny, M. : BIRCH: An Efficient Data Clustering Method for Very Large Databases. In: Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pp. 103–114. ACM Press, Canada (1996)
- [22] Kaur, S., Bhatnagar, V., Mehta, S., Kapoor, S. : Concept Drift in Unlabeled Data Stream. Technical Report, University of Delhi (2009)
- [23] Rousseeuw, P. J. : Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 53–65 (1987)
- [24] Spinoso, E. J., Ponce de Leon Ferreira de Carvalho, A., Gama, J. : OLINDDA: a clusterbased approach for detecting novelty and concept drift in data streams. In: Proceedings of the 22nd Annual ACM Symposium of Applied Computing. SAC, vol. 1, pp. 318–332. ACM, Seoul, South Korea (2007)
- [25] Fanizzi, N., d'Amato, C., Esposito, F. : Conceptual clustering and Its Application to Concept Drift and Novelty Detection. In: Proceedings of the 5th European semantic web conference on the semantic web (ESWC). LNCS, vol. 5021, pp. 448–452. Springer-Verlag, Tenerife, Spain (2008)