A new segmentation approach using dynamic variables on individuals

Nicolás Prieto
Master Student of Data Science and Analytics,
Universidad EAFIT, Medellín, Colombia,
nprieto@eafit.edu.co

Henry Laniado
Department of Mathematical Sciences,
Universidad EAFIT, Medellín, Colombia,
hlaniado@eafit.edu.co

Juan Carlos Monroy
Department of Marketing,
Universidad EAFIT, Medellín, Colombia,
jmonroyo@eafit.edu.co

March 13, 2021

Abstract

The problem of dynamic segmentation consists in finding the segments in which the individuals of a population must be grouped in different periods of time, considering that, as time passes, the general characteristics of the population will be changing, and therefore the segments must be evolving through time. In the literature, diverse techniques have been proposed to achieve the effect of obtaining segments that change through time. However, these techniques have not focused on balancing correctly the importance between past and present information. In this study, a new dynamic segmentation technique is proposed, which observes past behaviors to weight the importance of the variables in a clustering technique and uses the real values of the observations for the current period to find the clusters and to obtain a better segmentation. In addition, an alternative robust version of the proposed dynamic segmentation technique is presented, where some changes are made to have a better performance in presence of case-wise and cell-wise outliers. The performance of both proposals introduced in this work are compared with classical and recent techniques introduced in the literature, in simulated and real datasets with and without outliers. Results show that, for data without outliers, the proposed non-robust dynamic segmentation technique usually outperforms the other dynamic segmentation techniques, but when outliers are included in the dataset, the robust version of the proposed dynamic segmentation technique outperforms both the non-robust version and the other techniques presented in the literature.
Keywords: Clustering, dynamic segmentation, robust techniques, outlier treatment, data science.

1 Introduction

The problem of segmentation, proposed in Smith (1956), is well known in the literature. It involves the necessity to determine which are the possible sets in which the members of a population can be grouped. Segmentation presents great utilities because it allows analysts, scientists and businessmen to identify common patterns in certain groups of individuals and therefore focus on

specific parts of the population to be able to provide personalized treatments for each of these groups.

Throughout the years, many authors have conducted studies on segmentation approaches. For example, Haley (1968) suggested a segmentation approach that attempts to find causal relationships between the characteristics of the population and their future behavior. In addition, Vollering (1984) considered that the interactions between the population should be a key factor for determining the groups. On the other hand, Sharma & Lambert (1994) proposed a segmentation based on customer service needs, since each group may have different customer service requirements.

Cluster-based techniques can also be used for market segmentation, which can be either hierarchical, non-hierarchical or a combination of both, such as the approaches proposed in Punj (1983) and in Helsen & Green (1991). In more recent years, advanced machine learning techniques have been used for segmentation, such as artificial neural networks as mentioned in Kuo *et al.* (2002) or support vector machines as the approach proposed in Huang *et al.* (2007).

However, the general characteristics of the population may change through time, which leads to the necessity that the segments must be dynamic instead of static, i.e., they must change depending on the time. This property is explained in Blocker & Flint (2007), which mentioned that segments can be unstable because the individuals needs and characteristics may change through time and this should be reflected in changes in segment structures over time. Thus, there exists a recent interest in proposing techniques that are able to find the most appropriate groups for each period to improve the segmentation results, as the one proposed in Crespo & Weber (2005), in which they adapt the fuzzy c-means technique to be able to apply it in dynamic customer segmentation.

More recent approaches of dynamic segmentation include, for example, Benítez *et al.* (2014), in which each individual is represented by a multivariate time series consisting in all its historic and present values. On the other hand, Pereira & Mendes-Moreira (2016) used clustering techniques to find the best segments for each period independently: the technique is ran again from zero for each period, therefore using only the current values of each period to segment. Also, there are other alternative approaches as the one presented in Zhang *et al.* (2019), in which network clustering is analyzed; this approach uses information of the current and the previous period to find the segments.

The existing techniques of dynamic segmentation in the literature do not give enough focus on trying to achieve a balance between past and present data to improve their results. This balance implies that it is necessary to consider the behaviors of the segments in the past, but it is also very important to observe the characteristics of the individuals in the present to be able to obtain the best segmentation. This gap detected in the literature is the one that this research is expecting to fill: we propose a technique that weights the importance of the variables used in the segmentation according to the relevance of each variable in past periods, but it also gives enough importance to the real values of the observations in the current period to find the best possible segments, achieving therefore a balance in the use of past and present information.

It can also be seen in the literature that some of the existing dynamic segmentation techniques present a poor performance in presence of outliers in the dataset, which is caused due to the fact that these dynamic approaches usually are based on underlying clustering techniques that are not robust to outliers, such as it is mentioned in Munusamy & Murugesan (2020). Therefore, there

also exists a need to propose a dynamic segmentation technique that should be robust in the sense that its performance should not be greatly affected when there are outliers. Therefore, we aim to propose a modification for our technique to make it robust and allow it to perform well in presence of both case-wise and cell-wise outliers, which consist in contaminations of entire observations or only some variables of a set of observations respectively. For a further explanation of these types of outliers, readers should consult for instance Velasco *et al.* (2020) and the references therein.

In addition, to justify the utility of the creation of a new dynamic segmentation for the industry, many applications of segmentation were found in which a time factor is included. For example, EurekaFacts[1] applies segmentation techniques on the Hispanic market of the United States. They use socio-demographic information to cluster the territorial divisions of the country in 11 groups and therefore characterize the Hispanic population of each zone (the information is collected from Census Data available, keeping in mind that socio-demographic variables change over time). Moreover, having this segmentation defined, it is possible to dynamically profile customers according to the characteristics of the groups.

Many other example applications of dynamic segmentation exist in the literature. Cheng & Chen (2009) applied the known RFM model to segment the customers of a Taiwanese company that manages transactional data, which is a common approach for this type of data. On the other hand, Munusamy & Murugesan (2020) employed dynamic segmentation to find groups of customers for a retail supermarket in India based on their transactions. Also, Rezaee *et al.* (2018) used dynamic clustering on companies that belong to the Tehran Stock Market. In addition, Benítez *et al.* (2014) applied dynamic segmentation on the load profiles of energy consumption from Spanish customers. At last, Zhang *et al.* (2019) used dynamic community detection algorithms with Douban (the social network of China), and they identify the different communities formed through time.

Thus, the objective of this work is to introduce a segmentation approach that uses dynamic variables on individuals, which focuses on achieving a balance in the use of past and present information to obtain a better dynamic segmentation. In addition, this study also describes the modifications that we made to our proposed technique to be able to have a robust performance in presence of outliers.

This paper is organized as follows: in section 2 we describe some of the different existing techniques for static and dynamic segmentation that we implemented to use them as comparison points for our experiments. Section 3 introduces our main contribution in this work, which is our proposed segmentation based on dynamic variables on individuals, and it also explains the modifications that we made to it in order to create an alternative robust version of this proposed technique. Then, section 4 explains the performance measures that are used in our experiments to decide which technique is the best. In section 5, we present the experiments conducted both on simulated and real datasets, in which the performance of our proposed techniques and of the implemented existent approaches are compared and analyzed. Finally, section 6 contains the conclusions of this study and proposes future works in this topic.

[1] `https://www.eurekafacts.com/products/segmentos/`

2 Existing approaches

In this section, we present the description of the static and dynamic segmentation techniques existent in the literature that were implemented to compare them with our proposed techniques.

2.1 Static segmentation

The segmentation techniques in which their segments do not change through different periods of time can be considered static segmentation techniques. It is important to note that individuals may change their characteristics through time, but if the segments used to group them stay the same, the technique would be considered static.

A good example of this behavior could be the use of age groups to segment customers. If the age groups used to segment customers are fixed, then it may be possible that customers can change between one age group and another, but due to the fact that the ranges of age groups are fixed, the technique would therefore be considered static.

For the experiments shown in this study, a k-means algorithm using just data from the first period is used as the static technique (SKMP1). The algorithm only considers the data in the first period to fit the technique and find the centroids, and in the rest of the periods the new datapoints are assigned to the cluster of the closest centroid but the centroids are never readjusted, therefore having static segments.

2.2 Fixed segmentation

We decided to also include an approach similar to the previous one but with a modification. In this approach, k-means is also applied only for the first period, but in this case, individuals are not reassigned to the closest centroids in each period. Instead, each observation is considered to belong always to the same group that it was assigned in the first period, having therefore always the same individuals in each cluster for every period.

Therefore, in this approach, the variables that characterize a segment (for example, the mean values of each variable for all of their observations) would change through time, because the observations that belong to each of them evolve through time. However, due to the fact that in this case the technique is only using information from the first period to form the groups and the rest of the periods are not used by the approach, we decided not to classify it as a dynamic segmentation technique.

This approach can be therefore considered as a fixed k-means based on data from period 1 (FKMP1). However, it is important to note that this approach requires that exactly the same individuals are present in each of the periods, which is something that many times does not happen (for example, in customer segmentation, usually some new customers appear and some old customers leave in each period). This final property makes it more limited than other segmentation techniques (it cannot be applied on as many datasets as the others).

2.3 Existing dynamic segmentation techniques

In the literature, there exist some techniques that can be used for dynamic segmentation problems. Some of these techniques were implemented to compare their results with the proposed dynamic segmentation technique in both its non-robust and robust version. Even though some of these techniques were presented with a different base clustering technique, we used k-means as the base clustering technique for each approach to be able to have a fair comparison between them. The implemented dynamic segmentation techniques that can be found in the literature are the following:

- Benítez *et al.* (2014) used a dynamic segmentation technique based on time series clustering, in which each observation of the dataset is considered as a time series, where each period represents a point of it (TSKM). In the multivariate case, each point would be a vector, and therefore each individual would be represented as a time series of multivariate points, which can be seen as a $(m \times i)$ matrix, where m denotes the number of variables and i the number of periods up to the current one.

  This technique uses k-means on these time series by calculating the Euclidean distance between the observations, considering that each $(m \times i)$ can be represented as a one-dimensional vector of $m * i$ values, and computing the distance between these vectors. Due to this condition, this approach also requires that the same individuals are present in all of the groups and that there are no missing variables for any individual. However, as said before, many times this does not happen, which also leads to the fact that this technique cannot be applied on as many datasets as some of the other techniques.

- Pereira & Mendes-Moreira (2016) proposed the use of a clustering technique in each period to find the segments (IKM). In this approach, each period is considered completely independent to the others: the technique only involves applying a clustering technique for each period. In this case, past information does not influence in any way the segments for each current period.

- Munusamy & Murugesan (2020) mentioned the possibility of a dynamic segmentation technique in which, for each period i, all of the datapoints from periods 0 to i are clubbed together (CKM). The clustering technique is then applied on this clubbed data, and only the datapoints corresponding to period i are then observed to verify to which cluster each of them was assigned, obtaining thus different segments for the datapoints of each period. In this case, even though past and present information is used, each past period has the same importance as the current period when finding the segments, which is not a desired behavior.

3 New dynamic segmentation technique

In this section we present the principal contribution of this work: we explain the dynamic segmentation technique that we propose to obtain better results by balancing the importance of past and present information. We also describe a modification to the proposed dynamic segmentation technique to make it more robust to outliers.

3.1 Segmentation using dynamic variables on individuals

Such as it could be seen, existing techniques in the literature do not focus on balancing the importance of present and past information. They either consider each past period as having

the same importance as the current one when finding the segments (which is not correct because the current period should have a higher importance than each individual period of the past) or ignore the past information (which is also not correct because it can help bring stability and better performance to the method).

Therefore, a segmentation using dynamic variables on individuals (SDVI) is proposed, in which past information affects the weight of the variables used in the clustering technique, while the real values of the observations in the current period are weighted and used as the observations to be clustered. This allows the historic information to have a clear impact in the method because it determines which are the most important variables for the weighting, but the current period plays a very important role (bigger than each individual past period) because it contains the real observations being clustered in each period. Thus, this new approach can provide a better balance between the importance of past information, existent in the weight of the variables, and current information, existent in the real values of the observations being clustered in the current period.

To explain in more detail the proposed dynamic segmentation technique, the meaning of the used notation is specified as follows:

- X_i^t: Variable i in time t
- $C = \{c_1, c_2, \dots c_k\}$ set of clusters, note that the cardinality of the set is represented as $|C| = k$
- w_{is} : importance of variable i for cluster s
- W_i^t : importance of the variable i across all clusters in time t

The proposed dynamic segmentation technique starts by building a clustering of the first period with a k-means technique by using only the datapoints corresponding to that first period (in this implementation, the k-means technique used recalculates the centroid after each datapoint is reassigned and the Euclidean distance is used).

After finding the clusters for the first period with k-means, the importance that each variable had for each cluster (w_{is}) is calculated by using a supervised classification technique (the classifier should predict whether each datapoint belongs or not to each cluster). This supervised technique therefore should yield the importance of each variable for each cluster w_{is}. In this study, for the computational implementation, a technique based on randomized decision trees is used to find the importance of the variables, but other techniques could be considered. The mean importance of each variable across all clusters is stored as the importance of the variable for that period (W_i^t).

Next, the importances of the variables that are found in the first period are used to weight the dataset that is used in the second period to find the clusters, therefore using a weighted k-means for the second period, weighting with W_i^t each variable X_i^t (for the periods that occur after the second one, the mean of the importance of the variables (W_i^t) in all the previous periods is considered to weight the dataset). It is also important to mention that for each period $t > 1$, the centroids and the labels used by the k-means technique are initialized with the final centroids and labels found during period $t - 1$, this is done to keep some stability and momentum in the clusters through time and can help to obtain more consistent results.

The weighted k-means is therefore used to find the segments for each period t, but also an additional unweighted k-means is used in each period to obtain the importance of the variable for that period and to store the weights W_i^t for them to be used by the weighted k-means technique in the following period. The reason why an unweighted k-means is used to find the importance of the variables in each period is to prevent the weights found in the previous periods from having a great influence in the importance that is calculated for each variable in each period.

The previous steps are repeated for the observations of each period until all of the datapoints are assigned to a cluster. Figure 1 shows a diagram that explains visually the process followed by the proposed dynamic segmentation technique.

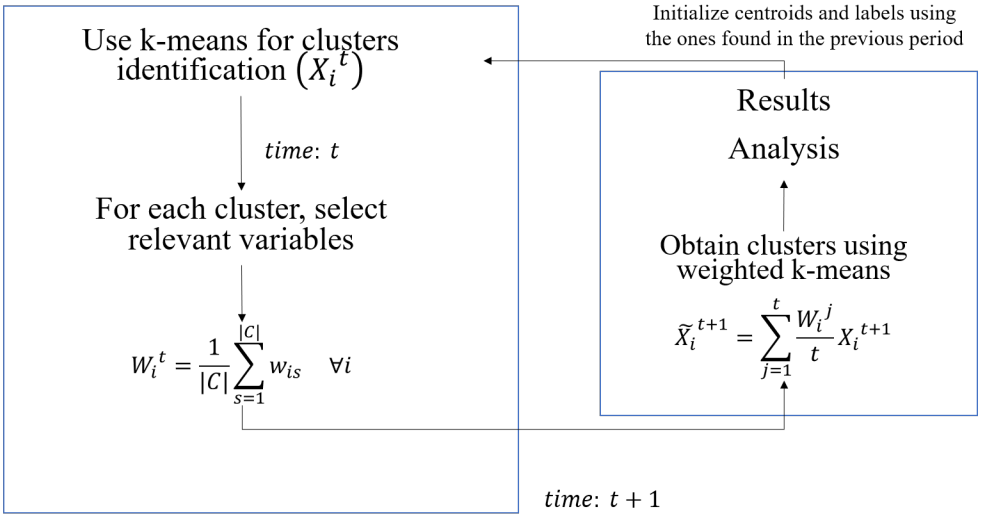

Figure 1: Proposed dynamic segmentation technique

3.2 Robust version of dynamic segmentation

Even though the proposed dynamic segmentation technique can exhibit a good performance (such as it is shown in the computational experiments), it is not designed to perform well in the presence of outlier observations, which makes it a non-robust technique. Therefore, an alternative robust version of the segmentation using dynamic variables on individuals (RSDVI) was designed, which allowed it to have a good performance in datasets in which outliers are present.

To create a robust version of the dynamic segmentation technique, some changes were made in its components. The first change that was made was that the k-means used by the technique in each of the periods (with and without weighted data) was changed for a k-medians. K-medians is an alternative version of k-means, in which instead of using the mean of each variable for calculating the centroid of its group, the median is used. The k-medians technique has been known for being more robust to outliers than the k-means, such as it is mentioned in Angelin & Geetha (2020).

The second change that was made to the technique to make it more robust consists in changing the distance used by the technique (Euclidean distance) for a robust function based on a modification applied to this distance. To accomplish this, we propose to trim the Euclidean distance, this is done

by considering only the 90% of the components that have lower square difference in the calculation (therefore, the 10% upper square differences are trimmed). Although this threshold was fixed in that value for this study, it could be modified in future applications of this technique.

For a better understanding of this robust function, consider the definition of the Euclidean distance (d_{euc}) between two vectors $X, Y \in \mathbb{R}^p$:

$$d_{euc}(X,Y) = \sqrt{\sum_{i=1}^{p}(X_i - Y_i)^2}$$

Note that, when calculating the Euclidean distance, a vector $C \in \mathbb{R}^p$ is implicitly calculated, where each component corresponds to $C_i = (X_i - Y_i)^2$. Now, it is possible to consider just the 90% of the components of vector C (rounded to the lowest integer) in which their values are the lowest for the calculation, that means a new vector will be created such as:

$$C_i^{trim} = \begin{cases} C_i & if \quad C_i \leq c_{90\%} \\ 0 & Otherwise \end{cases}$$

where $c_{90\%}$ corresponds to the 90th percentile of the values of the components from vector C.

Having the vector C^{trim}, it is possible to apply the proposed robust function (f_{re}) such as:

$$f_{re}(X,Y) = \sqrt{\sum_{i=1}^{p} C_i^{trim}}$$

This use of a robust function should be useful in the cases in which only some of the variables of a datapoint can be considered outliers.

In order to have a better understanding of this robust function, an application of it can be seen in the following example:

Consider two pairs of vectors (X, Y) and (X_{cont}, Y), in which we assume that the vector X_{cont} is a contaminated version of vector X where the ninth component represents an outlier value.

$$\begin{aligned} X &= [1, 0.5, 0.8, 0.9, 0.1, 0.2, 0.3, 0.7, 0.4, 0.3] \\ X_{cont} &= [1, 0.5, 0.8, 0.9, 0.1, 0.2, 0.3, 0.7, 10.4, 0.3] \\ Y &= [0.9, 0.4, 0.9, 0.7, 0.1, 0.1, 0.2, 0.8, 0.3, 0.6] \end{aligned}$$

The calculation of the Euclidean distance, as well as the value of the robust function for each pair of vectors can be seen in Table 1.

Table 1: Values of Euclidean distance and robust function applied to pairs of vectors (X, Y) and (X_{cont}, Y)

	(X, Y)	(X_{cont}, Y)
Euclidean distance	0.447	10.109
Robust function	0.332	0.436

Note that the Euclidean distance is affected to a great extent by the outlier component, since there is a large difference on the distance calculation when there are not outliers in the vector in comparison to the contaminated one (it changes from 0.447 to 10.109). It was expected to have a high difference in the Euclidean distance between the two pairs due to the fact that vector X is contaminated in one of its components, and this outlier has a great effect in the calculation of the Euclidean distance as shown in this example.

In contrast, when analyzing the result given by the robust function, it can be seen that it is more stable on its value in presence of outliers, since there is not a big difference between the value obtained using the non-contaminated vector against using the contaminated one (the value changes from 0.332 to 0.436). The difference between these two values is therefore much lower than the one obtained when using the Euclidean distance, which makes sense since vector X_{cont} is just a version of X that was contaminated in only one component. This function is able to achieve this lower value because the square difference between the ninth components, where the outlier is present, is set to zero during the calculation, which allows it to have a more robust behavior.

The third change that was made in the technique is that, instead of using the mean of the importance of the variables in all the periods (W_i^t), the median importance is used. This change can help when there are some periods that are very different to the rest: these periods can find importances of variables that are not consistent to the ones found in other periods, but by using the median importance instead of the mean, the effect of these strange periods in the weighting is minimized.

The robust version of the dynamic segmentation technique that is presented in the results section includes these three changes, and its performance is compared with the non-robust version of the dynamic segmentation and with other state of the art techniques.

4 Performance Measures

To assess the performance of the dynamic segmentation techniques under study, a measure of performance needs to be established. To achieve this, the real reference observation labels of the datasets must be compared with the found cluster labels of each segmentation technique (related to their corresponding group).

Considering that clustering techniques provide some labels related to the unsupervised learning process to represent groups found in the dataset, it cannot be expected that the labels given by the technique match with the real labels, that is why a correction-label process needs to be done in order to compare the predicted labels given by the clustering technique with the real labels. To make this correction, once all predicted labels are obtained by the clustering technique, the predicted centroids of each group are calculated (these centroids correspond to the mean of the variables of the observations in each predicted group). Then, since the real reference centroids are known, using the Euclidean distance the closest real centroid is found for each predicted centroid, and the predicted label for each cluster will be corrected by replacing it with the label that corresponds to its closest real centroid.

Now, by associating the real reference labels with the found cluster labels, it is possible to consider a confusion matrix of the dataset (a square matrix, where its dimensions depend on the number of classes under consideration) that summarizes how correct was the assignment of the labels in the segmentation technique. Each row of the matrix corresponds to the predicted class of the technique while each column represents the real class. Keeping in mind this representation of the information, the elements of the diagonal of the confusion matrix represent the number of correct observations that were assigned to their corresponding group (meaning a number of good predictions) whereas the elements outside of the diagonal correspond to errors in the classification assignment process, that means that a perfect confusion matrix would be a diagonal matrix (since the elements outside the diagonal will be 0), and that would mean the classification did not make any mistake.

Considering the confusion matrix, a first error metric can be defined. This metric corresponds to the Frobenius norm of the difference between the estimated confusion matrix and the perfect confusion matrix (the theoretical diagonal matrix). Let A be a matrix of dimensions $m \times n$, its Frobenius norm is given by equation (1).

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2} \tag{1}$$

Observe from the previous expression that, if the classification is perfect, then the norm is zero, while if some errors were made on the label assignment process, then this norm takes values greater than 0, therefore this norm is a suitable performance measure. It is important to note that this metric does not have an upper bound, but it still gives information to compare the results of predictions given by two or more segmentation techniques: when comparing the results, the technique that has a lower value on this metric corresponds to the one having the best performance.

A second metric also based on the perfect confusion matrix and the estimated confusion matrix is defined. Since the elements of the diagonal on a confusion matrix correspond to a well-performance indicator of the label assignment process, it is possible just to consider the diagonal of each matrix, which is a vector containing the number of datapoints that were assigned to their real corresponding cluster. Then, by calculating the Euclidean distance between the two diagonals, we can obtain a measure of error for this process (diagonal Euclidean distance). A value of 0 corresponds to a perfect label assignment process, but if this metric takes values greater than 0, some errors were made by the technique.

Moreover, some common metrics derived from the confusion matrix can be included to analyze the performance of the techniques under consideration (since the problem of verifying the correctness of the labels can be associated to a classification problem). In particular, in this paper the analysis is focused on the F1-Score measure (which is calculated based on two other metrics: precision and recall). The formulation of the F1-Score is presented in the following equation.

$$F1 - Score = 2 * \frac{precision * recall}{precision + recall} \tag{2}$$

where,

$$precision = \frac{\sum TruePositives}{\sum TruePositives + FalsePositives} \tag{3}$$

$$recall = \frac{\sum TruePositives}{\sum TruePositives + FalseNegatives} \tag{4}$$

The true positives, false positives and false negatives attributes can be extracted from the confusion matrix.

In other contexts, the F1-Score is also known as the F-Measure such as it is stated in Sasaki *et al.* (2007). This score is important because it provides a better index for analyzing the error (the basis of this measure is to have a balance between the information of precision and recall). The F1-Score measure in equation (2) is bounded by 0 and 1, where 1 is its best value (the expected one for having a perfect performance) and 0 its worst.

For the experiments shown in this study, the F1-score is calculated for all of the classes (clusters) considered in the technique. Then, the unweighted mean of the F1-score between all classes is calculated, and this is the value that is reported as the F1-Score (a macro-average F1-score). Due to the fact the F1-score tries to combine the effects of other metrics (precision and recall) and that it is bounded (which allows it to be easier to compare and validate), we consider it as the most important metric for our experiments. Therefore, it is the metric that we use to decide in case of non-concordant results between the different performance measures.

In addition, in the simulation experiments, after including outliers in the dataset, another metric is used in order to measure the sensitivity of the techniques under study in presence of contaminated data. This metric corresponds to the relative change on the F1-score metric in each scenario against the F1-score obtained in the scenario without outliers. The relative change of each technique m with the outliers percentage $out\%$ is defined as:

$$RC_m^{out\%} = \frac{|F1Score_m^0 - F1Score_m^{out\%}|}{F1Score_m^0} * 100\% \tag{5}$$

Where $F1Score_m^{out\%}$ is the F1-score metric of the technique m considering $out\%$ of outliers in the dataset.

At last, it is also important to mention that to calculate the global metrics for each experiment, the entirety of the dataset is considered to create the global confusion matrix of all the dataset, both for the real groups and for the groups found by each technique. The real confusion matrix would be a diagonal matrix, where each non-zero entry corresponds to the real total number of observations in each group through all periods (if some datapoints correspond to the same individual in different periods of time, they are counted apart for the confusion matrix).

5 Application in simulated and real datasets

This section presents experiments in real and simulated datasets to evaluate the performance of the proposed dynamic segmentation technique in its non-robust and robust version. Their performance

is compared with the other traditional static and dynamic segmentation techniques described in this work.

We implemented each of the techniques mentioned in this study in Python 3.7 for these experiments. Run times and performance measures for each technique in each experiment were calculated. The computer used to run these experiments has the following specifications: OS Windows 10, 32 GB RAM, 1TB HD, 6 cores and processor core i7-8750H.

For the computational experiments (considering random number generation and hyperparameters of the technique), we used a fixed random seed (to ensure replicability of the experiments), the number of iterations in each of the clustering techniques is 25, and to calculate the importance of the variables for the proposed technique we use the scikit-learn implementation of the Extra tree classifier model with its default architecture and we extract from it the attribute of feature importance of the variables.

The computational implementation of the technique, along with the experiments presented in this section can be found in the following GitHub repository: `https://github.com/psaldar/SDVI`.

5.1 Simulated dataset

5.1.1 Description of the simulated dataset

In order to perform computational experiments in which it is possible to observe the strength of the segmentation technique introduced in this study, it is necessary to simulate different groups (knowing the real label for each observation) for different time periods to apply the dynamic segmentation technique on them. Thus, the simulated dataset allows to make comparisons of the performance of different dynamic segmentation techniques (the predicted labels are compared with the real ones, and then performance measures are calculated to analyze their behavior).

The next steps describe the process on how the simulated dataset is created:

1. Define: the number of groups to be considered in the dataset (K), the number of periods (T), the number of observations in each period (N), as well as the number of variables (P) and the number of non-relevant variables (Q, where Q<P). Also a trend value (ϕ), a constant (θ), a maximum value of the standard deviation of each variable on the first period (S) and a proportion of individuals to preserve after each period (R) need to be established.

2. Create for each group to be simulated a random vector of dimension P to be considered as the center (or centroid) of the group in the first period (let C_i^t be the centroid of group i at time t), where each component of the vector is between $-V$ and V.

3. For each group, for each period, move every component of its center as a random walk with trend such as: $C_{ij}^t = C_{ij}^{t-1} + \phi + \epsilon_t$, where ϵ_t is white noise.

4. For each variable, for each group in the first period, a standard deviation is generated as an uniform random number between 0 and S (let σ_{ij}^t be a random vector containing the standard deviation of variable j of group i at time t).

5. For each group, for each period, move every component of its standard deviation vector as an absolute-value random walk such as: $\sigma_{ij}^t = |\sigma_{ij}^{t-1} + \epsilon_t|$, where ϵ_t is white noise.

6. Up to this step, a centroid (mean for the group) and standard deviation vector have been defined for each group and period. The following step corresponds to define how many observations are going to be generated for each group at each period (subject to the fact that the number of total observations across all periods must be equal to N). Define a lower and upper bound for the number of observations for each group, let $lowerB = \lfloor \frac{N}{2 * k} \rfloor$ and $upperB = \lfloor \frac{N}{k} \rfloor$ be those bounds, and let $Nobs_i^t$ be the number of observations of group i at period t. Then, for each period, for groups 1 to $k-1$ generate a random integer number between lowerB and upperB, for the kth group the number of observations corresponds to $N - \sum_{i=1}^{k-1} Nobs_i^t$ for each t. In addition, a proportion of the data of each group of each period will be preserved and be part of the population of the next period, in that case, the number of new observations to be generated for each period for each group ($Nobs_i^t$) will change to $Nobs_i^t - \lfloor Nobs_i^{t-1} \times R \rfloor$.

7. Then, randomly select Q variables that will have a low importance to categorize the groups (this is to simulate the fact that, in general, not all possible variables are important to segment a population). Let v be a vector of dimension P, where each entry is 1 unless its index corresponds to one of the Q selected variables, in that case its value in the vector will be 0. In addition, for those Q selected variables, for each group and for each period, their centers will correspond to 0 and their standard deviation will correspond to 1. Then, using this vector, it is possible to define a factor such as $fact = v \bigotimes Unif_{Nobs_i^t \times 1} \times \theta$ to be added to the simulation[2].

8. At last, to simulate the data to be used in the simulation study, for each period, for each group, the data is generated as follows:

$$Data_i^t = np.random.normal(size = (Nobs_i^t, P)) \bigotimes \sigma_i^t + C_i^t + fact$$

Where *np.random.normal* corresponds to a Python function that generates a random normal standard sample with $Nobs_i^t$ observations and P variables. Keep in mind that, for the simulated data, a proportion R of observations from previous period are appended to the simulated data for each group.

5.1.2 Simulated dataset experiments

To test the performance of the proposed version of dynamic segmentation in both its robust and non-robust version, it must be applied on datasets with and without outliers. Therefore, for the experiments, in addition to using the simulated dataset described before, we also included two types of outliers (one at a time) to the simulated dataset: case-wise and cell-wise outliers, as recommended in Velasco *et al.* (2020). The percentage of outliers is varied between 1% and 9% and results are shown for each scenario.

Some of the traditional static and dynamic segmentation techniques and the proposed dynamic segmentation technique in both its non-robust and robust version were applied in these contaminated

[2]let $\bigotimes$ be a cell-wise product

simulated datasets to measure their performance. The described FKMP1 and TSKM techniques are not included in these experiments because they do not adapt to the fact that new individuals join the group in each period and others leave the dataset, which is something that happens in this case. To compute the performance measures in the case-wise scenario, outliers are ignored and only the rest of the observations are used for this calculation (this is because in case-wise outliers new outlier points are added to the dataset, while in cell-wise outliers real datapoints are modified to produce the contamination).

In the simulation process, the following parameters were used: $K = 3$, $T = 15$, $N = 600$, $P = 10$, $Q = 3$, $\phi = 1$, $\theta = 5$, $V = 8$, $S = 2$, $R = 20\%$. In addition, after simulating the data, and before executing the segmentation techniques, the entire dataset was standardized. Also, the X and Y axis of the plots that show the dataset and the clusters found by the segmentation techniques correspond respectively to the first and second components obtained by the principal component analysis of the entire dataset (with or without outliers depending on the case).

Figure 2 shows a visualization of the simulated dataset (before including outliers) that is used for these experiments. This plot shows how the groups evolve through time and it is clear that the majority of the values of the population in the X axis (first principal component) increase their value on this axis as time passes (move to the right), which leads to the necessity of having dynamic segmentation techniques whose clusters can change for each different period.

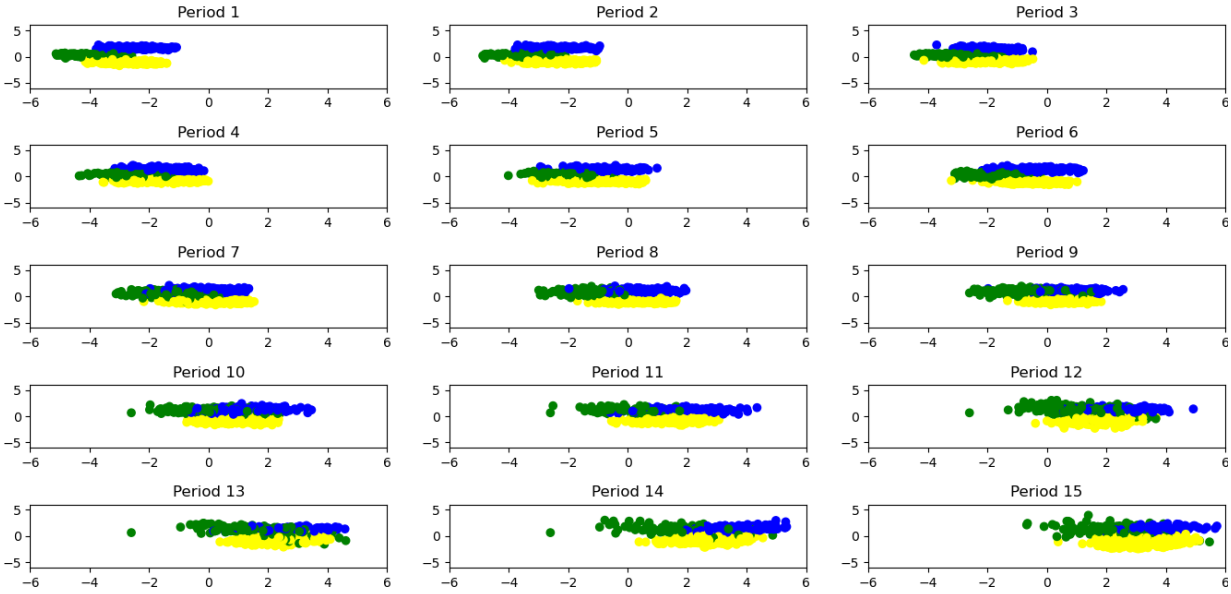

Figure 2: Simulated dataset used for dynamic segmentation examples. The first two principal components are plotted

5.1.3 Case-wise outliers experiment

Case-wise outliers (also called row-wise outliers) consist in specific observations in which all of their variables behave as outliers in comparison to the majority of the dataset. To incorporate case-wise outliers in the dataset, for each period a group of outlier observations is generated and appended

to the dataset.

The way in which this group of outliers is generated is the following: for each period, a random uniform vector where each component is a random uniform variable between 0 and 10 is generated, and each component will represent the standard deviation of each of the variables being considered. Also, another random uniform vector where each component is a random uniform variable between -50 and 50 is generated to represent the mean of each of the variables. Outlier observations are then generated randomly for each period from a normal distribution with these means and standard deviations and added to the dataset.

Figure 3 shows an example of the simulated dataset with 1% of outliers in each of the periods (the purple points are the added case-wise outliers):

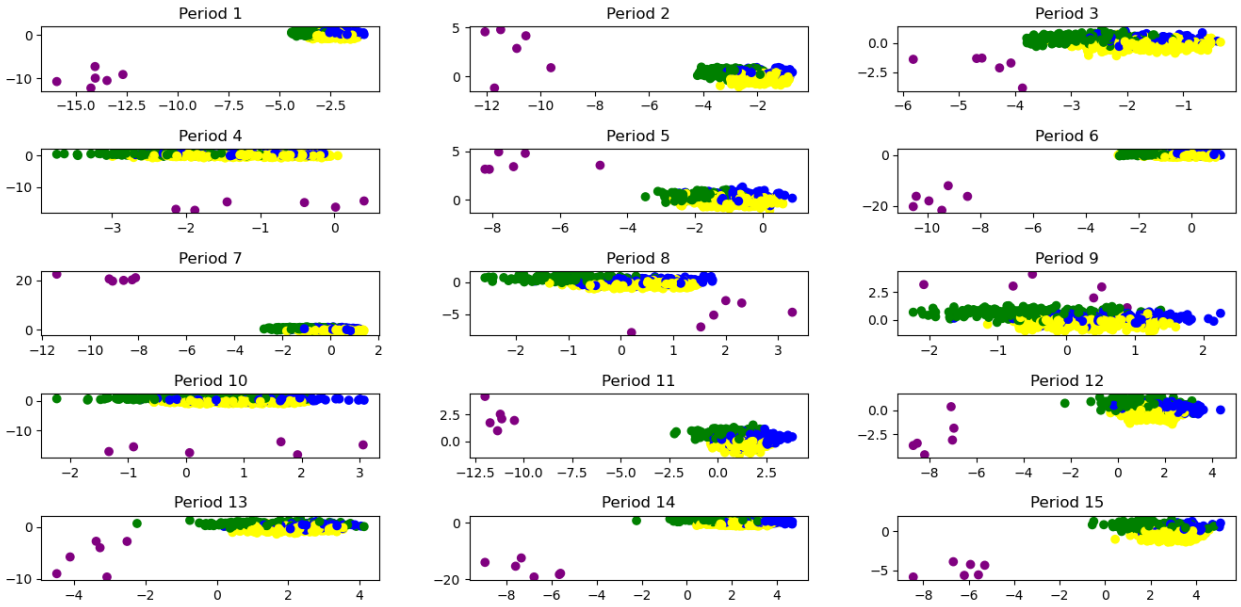

Figure 3: Contaminated dataset with 1% of case-wise outliers for each period. The first two principal components are plotted

Tables 2 and 3 show the Frobenius norm and F1-score metrics obtained by each of the segmentation techniques that were used in this experiment for each percentage of contamination:

Table 2: Performance by each technique on Frobenius norm including case-wise outliers

% outliers	SKMP1	IKM	CKM	SDVI	RSDVI
0%	1340.579	855.744	1653.445	122.744	638.832
1%	3203.564	1205.862	1956.122	2479.618	477.885
2%	2817.479	1880.890	3411.118	3356.081	524.414
3%	2878.716	1626.354	2931.994	3460.183	301.914
4%	2527.968	1856.706	3328.465	2577.830	826.800
5%	3488.633	1932.806	2842.393	3457.800	844.466
6%	2401.286	1856.992	3024.604	3342.913	1300.478
7%	2576.635	1874.296	3390.589	3635.026	303.664
8%	2757.499	1973.408	3438.614	3562.968	1167.142
9%	2557.893	1892.076	3948.994	3181.629	2245.275

Table 3: Performance by each technique on F1-Score including case-wise outliers

% outliers	SKMP1	IKM	CKM	SDVI	RSDVI
0%	0.818	0.891	0.763	0.981	0.892
1%	0.442	0.796	0.705	0.651	0.931
2%	0.557	0.688	0.456	0.493	0.936
3%	0.573	0.729	0.559	0.411	0.952
4%	0.612	0.692	0.487	0.588	0.891
5%	0.448	0.673	0.564	0.403	0.886
6%	0.626	0.686	0.526	0.415	0.824
7%	0.615	0.690	0.475	0.364	0.956
8%	0.557	0.668	0.418	0.389	0.853
9%	0.582	0.703	0.305	0.430	0.611

As it can be seen from the previous tables, when there are no case-wise outliers in the dataset (0% outliers), the proposed dynamic segmentation technique in its non-robust version outperforms its robust version and the other segmentation techniques of the literature in both F1-score and Frobenius norm, making it the best choice for a dataset without outliers.

However, for any percentage of outliers included in the dataset between 1% and 8%, it can be seen that the robust version of the proposed dynamic segmentation technique outperforms its non-robust version and all other techniques of the literature in both metrics, while, on the other hand, the non-robust version presents a poor performance when outliers are present. In addition, it can also be noted that, when there are no outliers, the robust version presents just a slight decrease in performance in comparison to the non-robust version, but its performance is still very good. Therefore, it is possible to conclude that the robust version of the technique performs well in datasets with and without outliers, making it a more stable and trustworthy technique than the non-robust version and the other analyzed segmentation techniques.

Table 4 shows the relative change in F1-score between the contaminated and non-contaminated dataset. It shows that the metric of the robust proposed technique is the one that is least affected when a percentage of outliers up to 8% is included, while the other techniques present higher losses

in performance in the presence of outliers (in particular, for the non-robust version of the proposed technique, these losses in performance are high).

Table 4: F1-Score - Relative change against performance without outliers

% outliers	SKMP1	IKM	CKM	SDVI	RSDVI
0%	0.0%	0.0%	0.0%	0.0%	0.0%
1%	46.0%	10.7%	7.6%	33.6%	4.3%
2%	31.9%	22.8%	40.2%	49.7%	4.9%
3%	30.0%	18.1%	26.8%	58.1%	6.7%
4%	25.2%	22.4%	36.1%	40.1%	0.1%
5%	45.3%	24.5%	26.1%	58.9%	0.7%
6%	23.5%	23.0%	31.0%	57.7%	7.6%
7%	24.8%	22.5%	37.8%	62.9%	7.2%
8%	32.0%	25.0%	45.2%	60.3%	4.3%
9%	28.9%	21.1%	60.0%	56.1%	31.5%

Figure 4 shows the evolution of the F1-score metric for each technique as more outliers are added to the dataset. The plot shows that, even though in the no-contamination case the non-robust version is the one with the best performance, for all other percentages of contamination between 1% and 8% the robust version is the one that presents better and more stable results, therefore making it a better approach. However, it is important to notice that in a percentage of outliers of 9% the robust proposed technique has a great loss of performance. This may indicate that in this percentage of outliers the robust technique reaches its breakdown point, which is defined in Rousseeuw (1984) as the smallest percentage of contaminated data that can cause the estimations to take on arbitrarily large aberrant values.

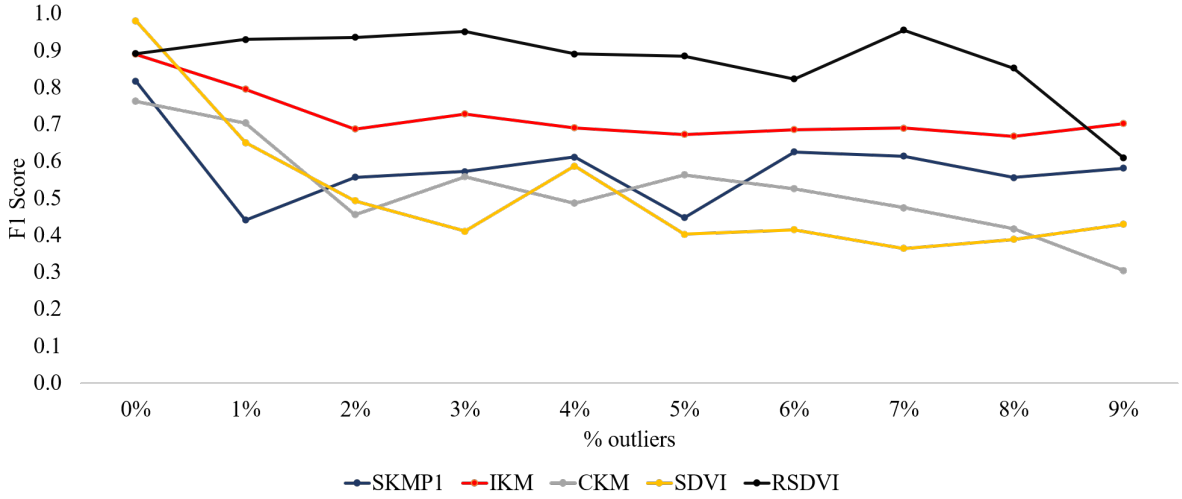

Figure 4: F1-Score performance in presence of outliers - case-wise scenario

Figure 5 shows the evolution of the relative changes for different contamination percentages. It shows that the results of the robust technique are in general more stable than those of the other

techniques, this means that the robust technique suffers a lower change in performance as more outliers are added in comparison to the other techniques.

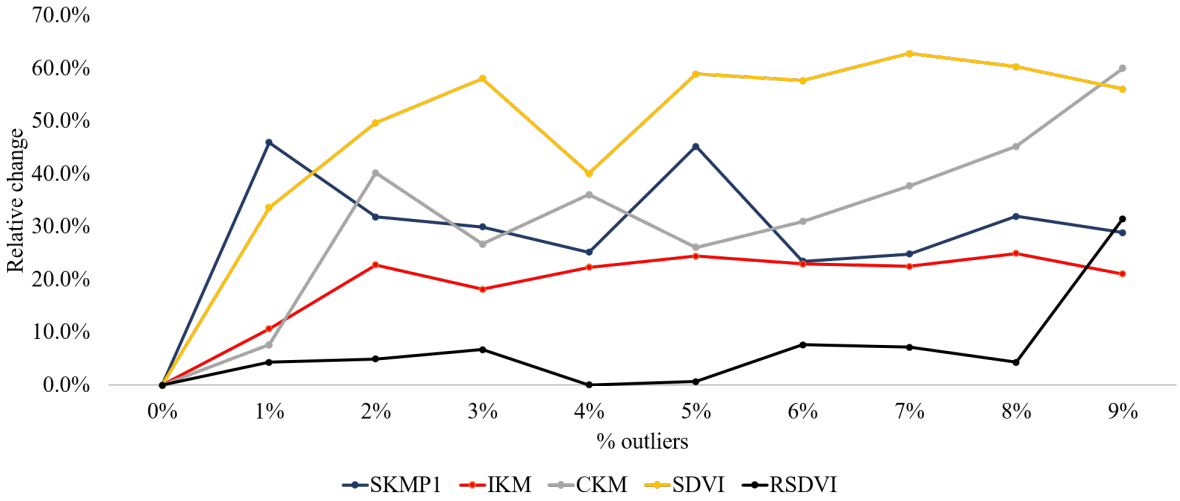

Figure 5: F1-Score relative change performance in presence of outliers - case-wise scenario

Next, we show the run times of each of the techniques in Table 5. These times show that the CKM approach has a very high computation time in comparison to the other techniques. The second highest time corresponds to the RSDVI; it is not very high for this dataset and thus does not represent a limitation in this scenario, but it is something that must be considered for larger datasets. SDVI has a run time of less than half than the one of RSDVI, this may be because computing the value of the robust function requires to compare all of the element-wise squared differences to exclude the highest one, which may be expensive. We also mention that IKM has a computation time of approximately half the time of SDVI, which makes sense because SDVI applies two times k-means per period (weighted and unweighted) while IKM applies it only once per period. Finally, SKMP1 has a very low run time, this is because it only applies k-means once in the first period.

Table 5: Median run times of each technique for the different contamination percentages: case-wise outliers

Technique	Execution time (seconds)
SKMP1	2.791
IKM	14.812
CKM	367.899
SDVI	31.749
RSDVI	70.041

Figure 6 shows the clusters found in the first four periods by the SDVI with 1% of case-wise outliers (centroids of the clusters are plotted as red crosses). It shows the problems that the non-robust version has: in the third period, the outliers are considered as one of the clusters, which is not the expected behavior. In addition, in period 4, the centroids were initialized with the final centroids of period 3 and one of them was of a group of outliers; this causes that in period 4 there are no datapoints assigned to one of the clusters (green cluster in the image), thus the technique finds only

two clusters and this affects in a great way its performance.

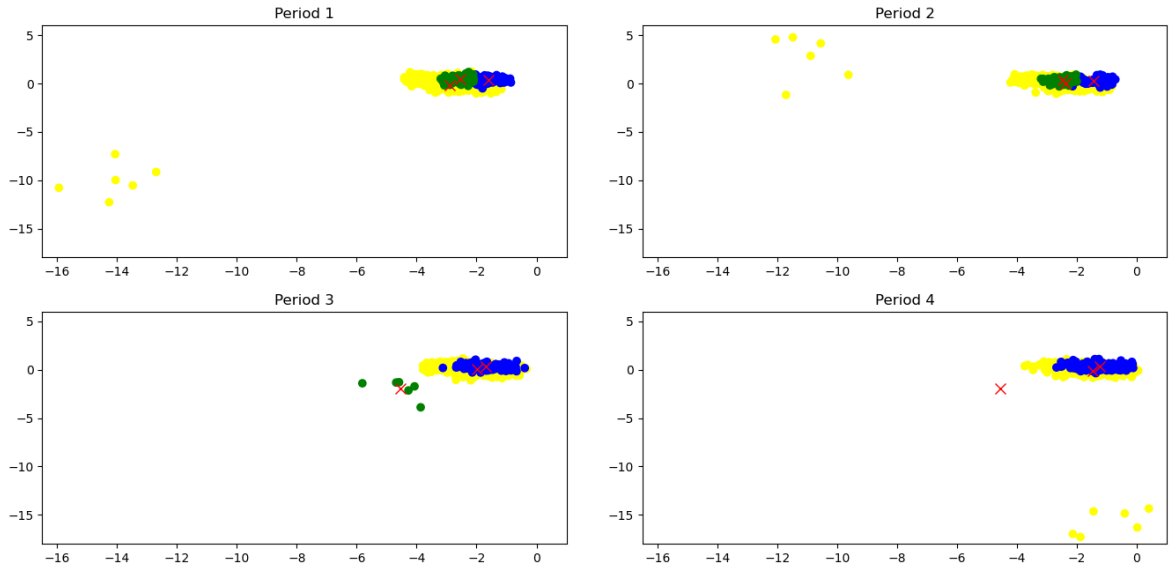

Figure 6: Clusters found with the non-robust proposed technique for the first 4 periods with 1% outliers. The first two principal components are plotted

Finally, Figure 7 shows the clusters found in the same periods by the RSDVI with 1% of case-wise outliers. In this scenario, it does not happen in any period that the outliers are considered as one of the clusters, and also the problem of having centroids without any datapoints assigned in a period does not occur. This explains why the performance of the robust technique is much better than the one of its non-robust counterpart.

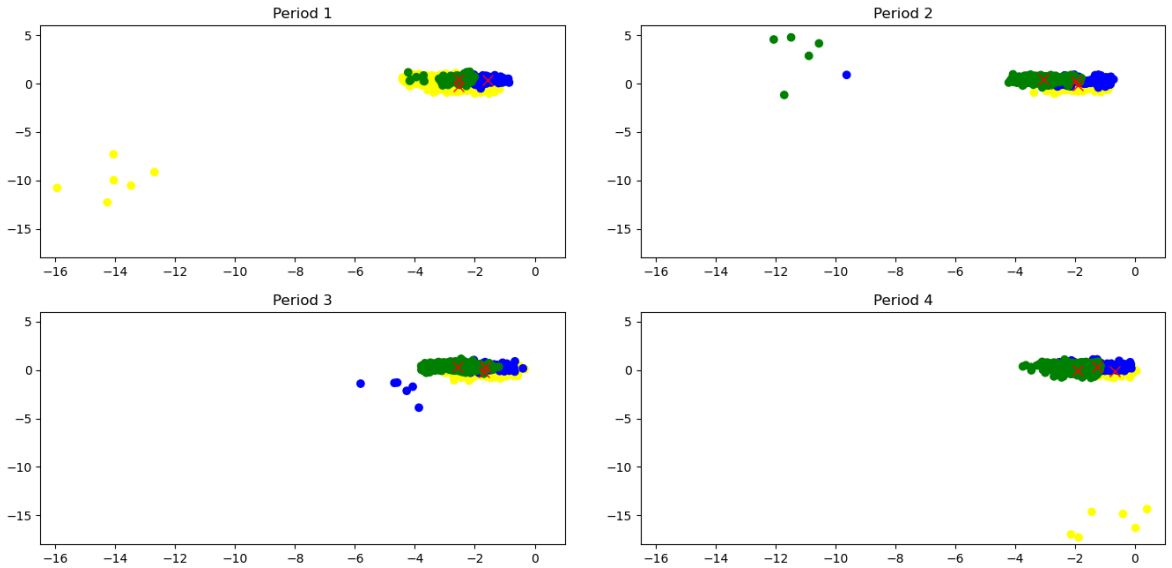

Figure 7: Clusters found with the robust proposed technique for the first 4 periods with 1% outliers. The first two principal components are plotted

Based on the results of this experiment, it can be seen that to be able to successfully apply dynamic

segmentation in a dataset which has case-wise outliers, the robust version of the segmentation approach using dynamic variables on individuals (RSDVI) should be used.

5.1.4 Cell-wise outliers experiment

Cell-wise outliers are a rare type of outliers that consist in having extreme values in just some of the cells of the data (those cells can correspond to different variables at different observations). In contrast to case-wise outliers, where an entire group of observations (rows) have different characteristics (i.e. having a different mean, variance, etc.), the cell-wise outliers therefore consider contamination in just some of the cells of the data matrix. This contamination consists in propagating the outliers such as it was stated in Alqallaf *et al.* (2009), where the cell-wise outliers correspond to a modification of the case-wise outliers, in which, instead of contaminating the entire row, only a proportion of their cells are contaminated. For a better understanding, let $X_{n\times p}$ be the data matrix, and each entry corresponds to the value of a variable for an observation (let x_{ij} be the value of variable j for observation i), where each value of the matrix is considered a cell (that means that there are $n \times p$ cells).

To incorporate cell-wise outliers, a percentage of cells (α) to be contaminated must be defined. After selecting the percentage of cells to be modified, the following transformation is applied:

$$x_{ij}^{cont} = x_{ij} + k * SD(X_j)$$

where k is a value from 1 to 10 as suggested by Velasco *et al.* (2020) and $SD(X_j)$ corresponds to the standard deviation of the variable X_j. Figure 8 shows an example of the simulated data including cell-wise outliers contaminating 3% of the cells.

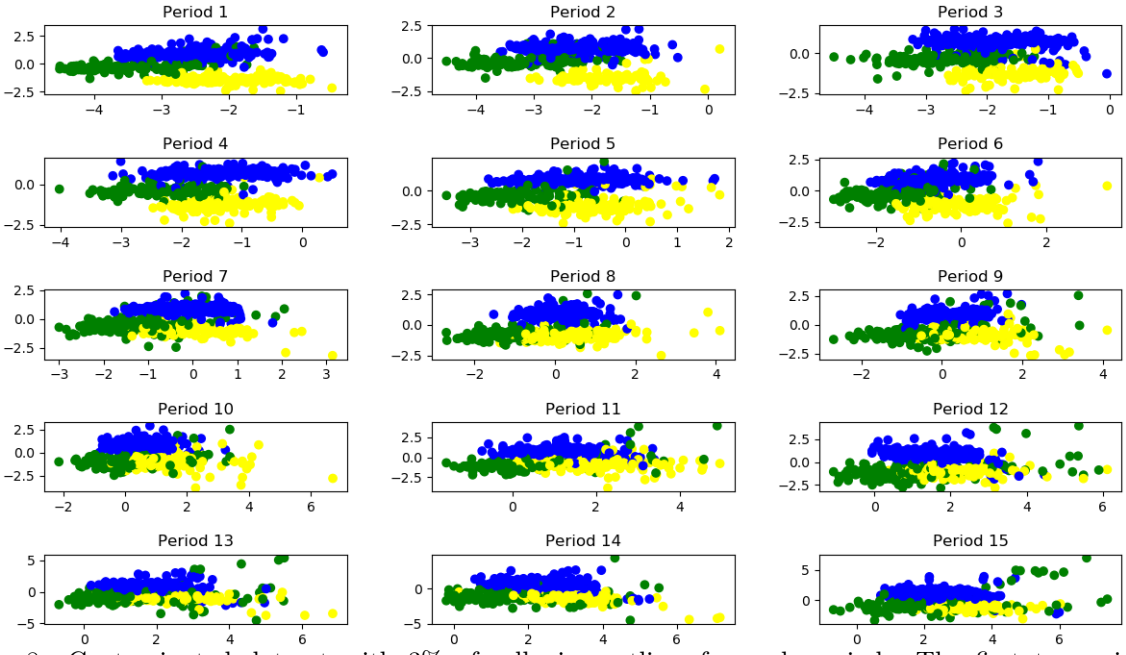

Figure 8: Contaminated dataset with 3% of cell-wise outliers for each period. The first two principal components are plotted

Such as it can be seen in the previous figure, keeping in mind that a multidimensional dataset is being mapped to its first and second principal components, the cell-wise outliers are not as visible in a plot as in the case-wise scenario, but as an intuition, some of the outliers may correspond to points that are further away from its corresponding group. This is because an extreme datapoint usually, but not always, corresponds to an extreme projection in principal component analysis.

Tables 6 and 7 correspond to the metrics obtained by each technique depending on the percentage of contaminated cells in the dataset.

Table 6: Performance by each technique on Frobenius norm including cell-wise outliers

% outliers	SKMP1	IKM	CKM	SDVI	RSDVI
0%	1340.579	855.744	1653.445	122.744	638.832
1%	1649.056	1348.579	2750.412	143.136	893.642
2%	1690.921	1269.354	2734.786	158.732	1257.027
3%	2989.462	2212.382	3695.765	2497.135	1113.170
4%	3647.195	2704.656	3765.164	4159.182	1179.461
5%	4083.746	2714.011	3723.698	4084.292	1204.761
6%	4033.372	3068.216	3491.084	4033.360	396.535
7%	4004.480	3048.986	3667.155	4004.529	356.592
8%	3974.183	3249.879	3631.730	3975.024	308.110
9%	3610.340	3591.029	3596.907	3942.576	402.201

Table 7: Performance by each technique on F1-Score including cell-wise outliers

% outliers	SKMP1	IKM	CKM	SDVI	RSDVI
0%	0.818	0.891	0.763	0.981	0.892
1%	0.780	0.766	0.583	0.978	0.896
2%	0.773	0.785	0.558	0.975	0.851
3%	0.485	0.643	0.394	0.570	0.869
4%	0.366	0.549	0.361	0.264	0.818
5%	0.281	0.533	0.369	0.281	0.811
6%	0.290	0.464	0.392	0.290	0.938
7%	0.293	0.467	0.376	0.293	0.942
8%	0.297	0.436	0.377	0.297	0.949
9%	0.368	0.359	0.380	0.301	0.934

Such as it is seen in the tables above, when there is no presence of outliers in the dataset, the non-robust proposed dynamic segmentation technique gives the best performance, while in second place comes its robust version (this analysis corresponds to the lower values given by the Frobenius norm as well as the higher values reported on the F1-score).

In addition, looking at the performance contaminating 1% and 2% of the cells, the SDVI and RSDVI keep reporting the best values (noting that the non-robust technique gives the best values, but the robust one still presents a good performance). However, after contaminating 3% or more of the cells, the non-robust version of the proposed technique cannot handle those extreme values and reports a very bad performance, whereas the robust version still keeps its good performance.

This means that the proposed technique in its non-robust version is very sensitive to the presence of cell-wise outliers, but its robust version is a good strategy to address this issue.

Moreover, to continue analyzing the performance of the techniques, Table 8 displays the relative change of the metric F1-score in presence of outliers against the metric obtained by each technique without outliers.

Table 8: F1-Score - Relative change against performance without cell-wise outliers

% outliers	SKMP1	IKM	CKM	SDVI	RSDVI
0%	0.0%	0.0%	0.0%	0.0%	0.0%
1%	4.7%	14.0%	23.6%	0.3%	0.4%
2%	5.5%	11.9%	26.9%	0.7%	4.5%
3%	40.7%	27.8%	48.3%	41.9%	2.5%
4%	55.2%	38.3%	52.8%	73.1%	8.3%
5%	65.6%	40.2%	51.6%	71.4%	9.1%
6%	64.6%	47.9%	48.6%	70.4%	5.2%
7%	64.2%	47.6%	50.8%	70.1%	5.6%
8%	63.7%	51.0%	50.6%	69.8%	6.4%
9%	55.0%	59.7%	50.3%	69.3%	4.8%

In order to state that the technique is robust and not sensitive, the expected behavior is to have a small value on this relative change (since the output of the technique in presence of outliers would be almost the same to its performance when outliers are not included). Such as it was mentioned before, the robust version of the proposed technique is the least sensitive to cell-wise outliers (that means the robust representation that was formulated addresses correctly the high sensitivity problem of the proposed technique). In addition, in Figures 9 and 10 the visual representation of the evolution of F1-score and its relative change is presented.

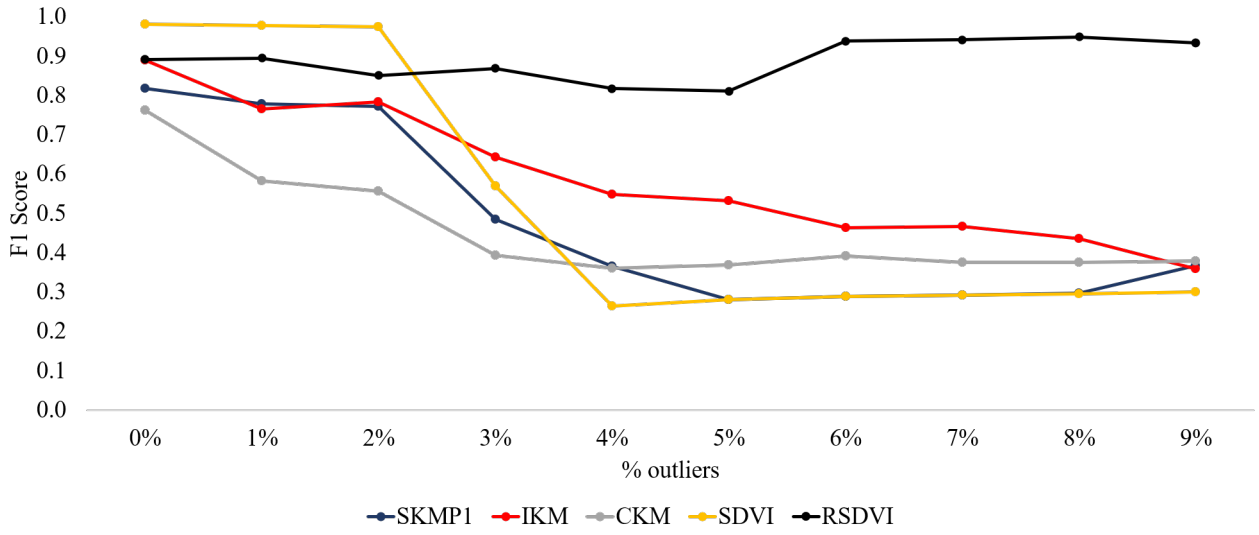

Figure 9: F1-Score performance in presence of outliers - cell-wise scenario

Figure 9 shows that, after contaminating cells, the performance of most of the techniques tends to

decrease, but the robust version of the proposed technique is very stable on its performance (suffers just low decreases in comparison to the other techniques).

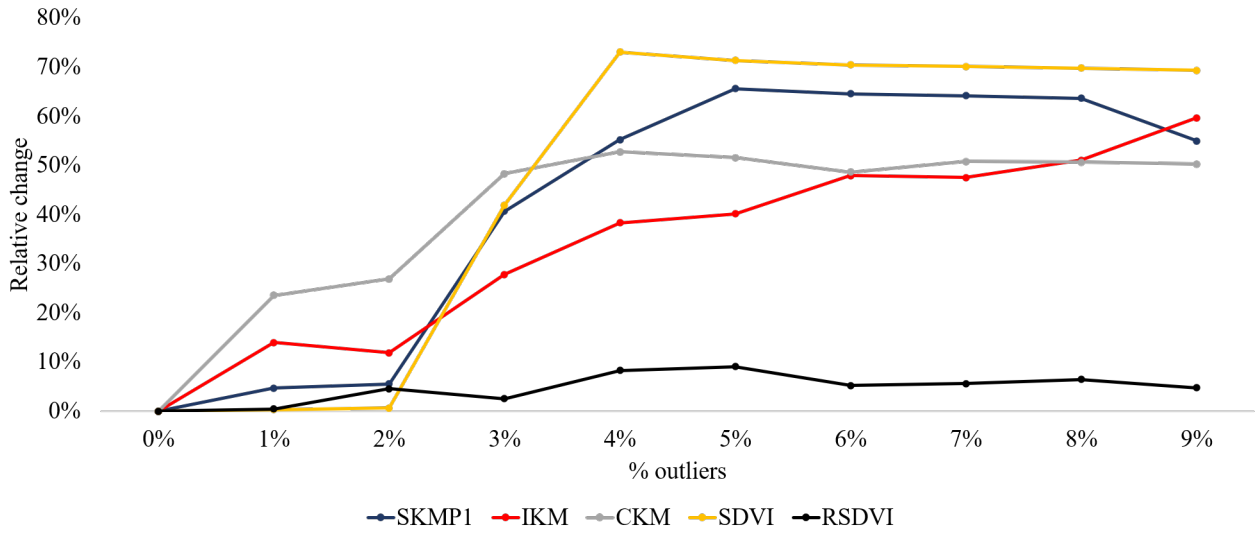

Figure 10: F1-Score relative change performance in presence of outliers - cell-wise scenario

In Figure 10, a breakdown point for most of the techniques can be seen at 3% contaminated cells. This breakdown point is identified because it is clear that almost all techniques (except the robust version of the proposed one) increase at a great rate their relative change when this threshold is reached.

Next, we show the execution times of each of the techniques in Table 9. These times are very similar to the ones presented for the case-wise scenario and the ranking between the techniques is the same. The only difference is that these median times may be slightly lower, but this happens because case-wise outliers are adding new observations to the dataset, while cell-wise outliers just modify existing observations (more observations lead to a higher computation time due to the clustering algorithm being used).

Table 9: Median run times of each technique for the different contamination percentages: cell-wise outliers

Technique	Execution time (seconds)
SKMP1	2.690
IKM	14.482
CKM	304.105
SDVI	28.314
RSDVI	66.551

Figure 11 shows the clusters found by the non-robust version of the proposed technique on the first 4 periods when 3% of the cells are contaminated (centroids of the clusters are plotted as red crosses). Since the first period it was able to identify 3 groups but the blue group seems to be formed by some outliers variables due to its low cardinality. Moreover, comparing the groups found by the non-robust version against the real ones in Figure 8, it is possible to observe that some groups were

not identified correctly, which explains the low overall performance on the tables presenting the metrics.

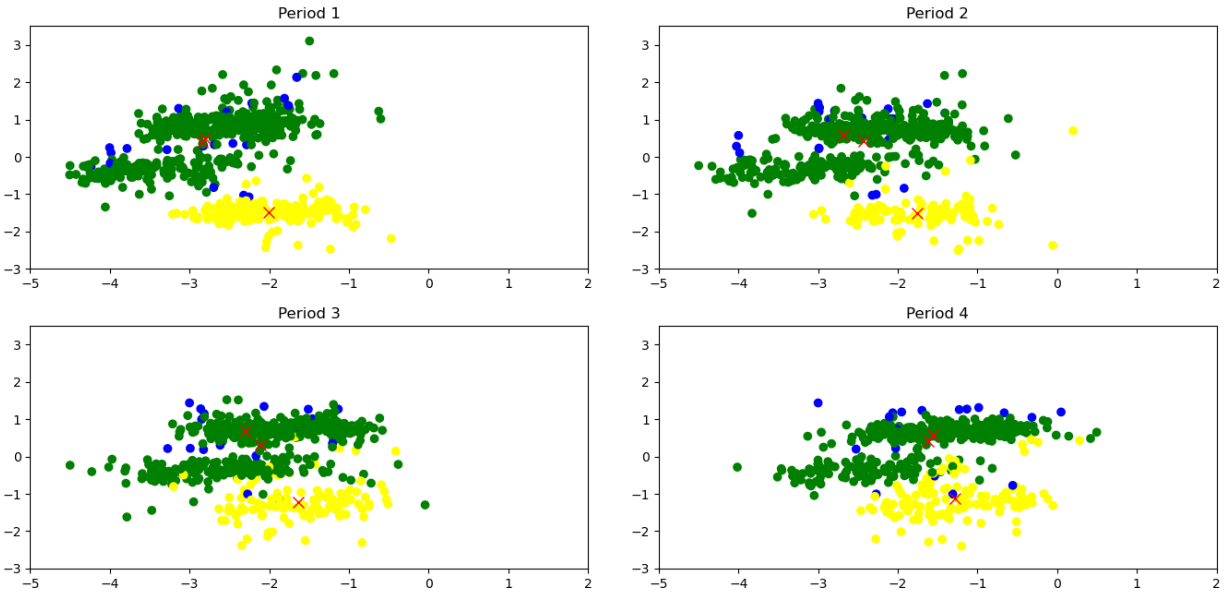

Figure 11: Clusters found with the non-robust proposed technique for the first 4 periods with 3% outliers. The first two principal components are plotted

It is important to consider that, although the datapoints of this blue group seem to be have a great separation from each other, the plot is showing only the first two principal components from the dataset. Therefore, even though some of these blue points seem to have a high dispersion and even in some cases they appear in the middle of the green group in the plot, this behavior can be explained due to the fact that only a couple of principal components are being shown. Cell-wise outliers usually contaminate only one or two variables per observation that is contaminated, which causes that sometimes it can occur that these contaminated variables do not play a great role in the principal components that are plotted and therefore these blue points may not look very close to each other in the plot, but they can still play a great role in the clustering technique because the technique considers all of the variables and not only the first two principal components to do the segmentation, which leads to them being grouped in the same cluster.

On the other hand, Figure 12 displays the output of the robust version of the proposed dynamic segmentation technique. It is possible to see that, for the four periods, the groups found are very similar to the real groups plotted in Figure 8, which is the desired behavior. Hence, with this visual representation, it is possible to understand why the performance of this robust version is much better than the one of its non-robust counterpart.

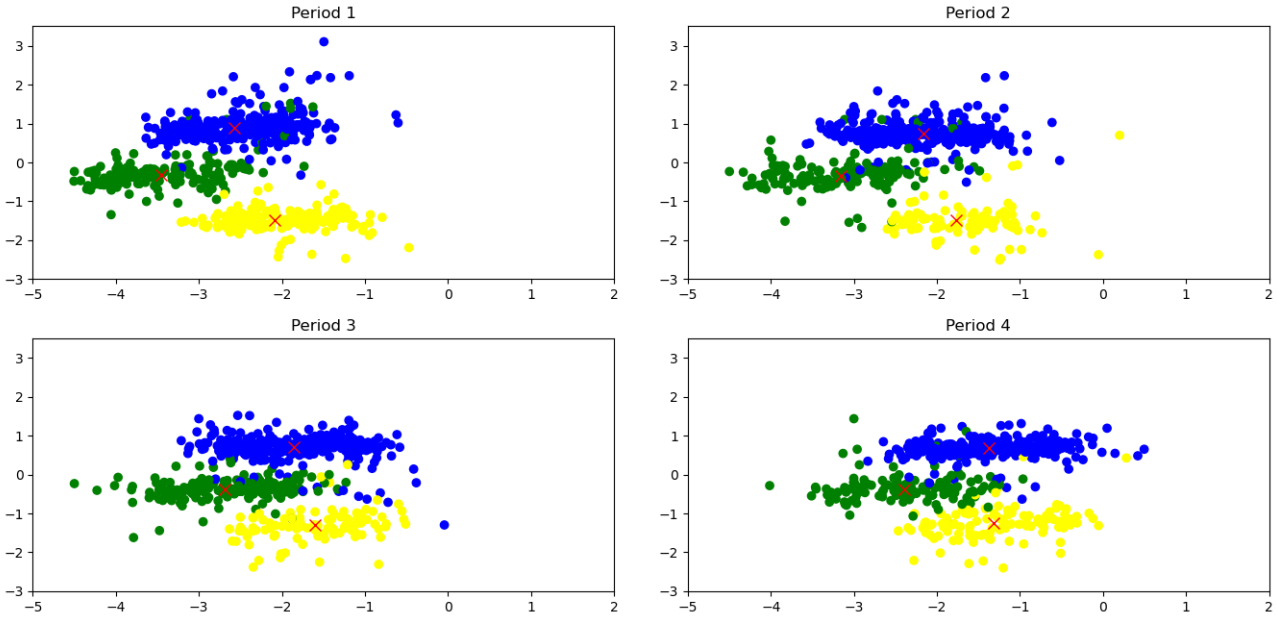

Figure 12: Clusters found with the robust proposed technique for the first 4 periods with 3% outliers. The first two principal components are plotted

Based on the results of this experiment, it can be seen that the robust version of the segmentation approach using dynamic variables on individuals (RSDVI) should be the one used when dynamic segmentation must be applied on a dataset with cell-wise outliers.

5.2 Real dataset

5.2.1 Description of the real dataset

Gapminder is an independent Swedish foundation that provides an open data source containing economic and socio-demographic information for most of the world countries for different periods of time, as well as other indicators related to energy, health, society and much more.

Considering that data found in Gapminder (2020) corresponds to indicators and variables of countries through time, it is useful to create different scenarios in which segments for countries can be determined for different periods of time with the use of dynamic segmentation techniques. Thus, we decided to use the real data of indicators of countries over time extracted from Gapminder to conduct experiments, in order to compare the performance of the non-robust and robust proposed techniques with the other implemented static and dynamic segmentation techniques from the literature. The described FKMP1 and TSKM techniques are also included here because in these experiments we consider the same individuals for each of the periods (the changing factor is the value of the variables of these individuals for each of the periods).

5.2.2 Experiments for Gapminder datasets

Three experiments are performed using the Gapminder data source to test the performance of the segmentation techniques in a real application. In each of the experiments, we consider the values

of different economic and socio-demographic variables for a set of countries for each period in a certain time frame. These variables are used by the clustering techniques to find the groups for each period, and these groups are then compared with a reference real classification of the countries (which is based on the level of development of each country as measured by the UNDP or the World Bank depending on the experiment) to calculate performance measures.

Table 10 shows the different variables from Gapminder that were included in each of the experiments:

Table 10: Gapminder variables used in each experiment

Variable	Experiment 1	Experiment 2	Experiment 3
population_density_per_square_km		X	X
population_growth_annual_percent		X	
population_total		X	X
exports_percent_of_gdp			X
gdp_per_capita_yearly_growth	X	X	
imports_percent_of_gdp			X
income_per_person_gdppercapita_ppp_inflation_adjusted	X	X	X
total_gdp_us_inflation_adjusted	X		X
child_mortality_0_5_year_olds_dying_per_1000_born	X	X	X
children_and_elderly_per_100_adults	X	X	X
children_per_woman_total_fertility	X	X	X
life_expectancy_years	X	X	X
internet_users	X		X
mean_years_in_school_men_25_to_34_years	X	X	
mean_years_in_school_women_25_to_34_years	X	X	

In addition, we clarify that the variables are standardized before being used by each of the clustering techniques. The standardization for each variable is applied on the entire dataset (the one that contains all of the observed values for the countries in all periods). Moreover, we mention that the X and Y axis of the plots that show the observations of the dataset and the clusters found by the segmentation techniques correspond respectively to the first and second components obtained by the principal component analysis of the entire dataset.

5.2.3 Experiment 1

For the first experiment, we decided to use as the real reference segments a three-group classification of countries according to their development level, which is based on a classification found in the Human Development Report published by the UNDP (2010). Based on it, we classify countries as developed, developing or least developed countries (LDC) depending on their development level.

To determine the countries that belong to the group of developed countries in each year, we use the Human Development Index (HDI) as described by the UNDP (2020). The HDI is an index that considers variables related to education, life expectancy and gross national income per capita to measure the level of development of a country. Data containing the HDI of each country was gathered from Gapminder for the periods between 1990 and 2018 and from the site of the UNDP (2015) for the periods between 1980 and 1989. Values for missing years are imputed using a linear regression of the values for each country. As explained in Nielsen (2011), UNDP defines that countries ranked over the 75 percentile of HDI are considered as the developed countries, therefore

we use this threshold to form this group.

In addition, the list of Least Developed Countries (LDC) by year is also gathered from the site of the UNDP (2018). This dataset includes a list of the countries belonging to the LDC category. This classification is reported every 3 years starting from 2000 and having the last report on 2018 (i.e., there is a classification for 2000, 2003, ..., 2018). We assume that between this 3-year lapse there is no change of status between the countries. This list is used to identify the least developed countries group for the real reference classification.

Finally, we consider that every country that does not belong to the developed or LDC category is considered as a developing country, forming therefore our third real reference group. It is important to mention that this group accounts for approximately 50% of the countries, which leads to the fact that there are unequal sizes between the clusters. These three groups are therefore considered to be the real reference classification labels.

For this experiment, based on the information available of the list of least developed countries, we decided to consider the periods between 2000 and 2015 as our dataset. The variables that we include in the dataset are variables that are all related with human development and correspond to 153 countries. These variables are shown in Table 10.

Thus, each segmentation technique was applied to the defined dataset for this experiment. Table 11 shows the performance measures calculated on the global confusion matrix (the one that contains the data of all periods) for each of the techniques:

Table 11: Overall metrics - Gapminder experiment 1

	SKMP1	**FKMP1**	**TSKM**	**IKM**	**CKM**	**SDVI**	**RSDVI**
Frobenius norm	701.330	408.754	417.097	468.468	443.193	413.422	495.542
Diagonal Euclidean distance	545.210	323.660	323.025	374.655	352.525	323.348	396.297
Accuracy	0.716	0.823	0.807	0.793	0.799	0.808	0.779
Precision	0.726	0.813	0.800	0.784	0.788	0.799	0.773
Recall	0.749	0.844	0.820	0.817	0.813	0.820	0.808
F1-Score	0.719	0.823	0.807	0.795	0.798	0.807	0.783

The previous table shows that our proposed technique in its non-robust version (SDVI) presents the second-best performance for this experiment, being topped only by the FKMP1 technique and having a performance very similar to TSKM. It is important to mention that the FKMP1 and TSKM approaches have the limiting factor of requiring that the same individuals appear in all of the periods, which is something that does not occur in many segmentation applications (for example, in customer segmentation usually different customers are present in each period), making them less flexible than our proposed technique.

In addition, as expected, the static segmentation technique (SKMP1) is outperformed by all other approaches. It is also important to note that, even though our robust version of the proposed technique (RSDVI) was outperformed, its metrics were still close to all other dynamic segmentation techniques, which indicates that it also had a decent performance. This may be because in this case there are not too many evident outliers in the dataset (as seen in Figure 13), therefore non-robust

techniques perform well, but a robust technique should also have a decent performance.

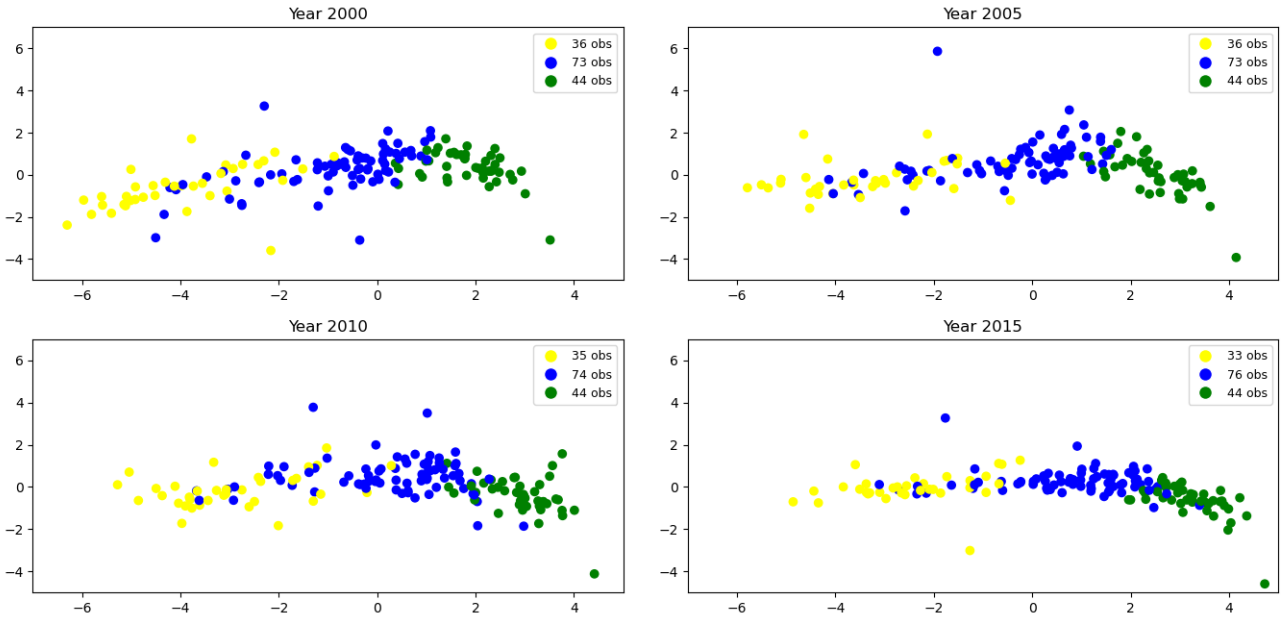

Figure 13: Reference classification groups for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

Now, we proceed to analyze the evolution of the performance of each technique through the different periods, as seen in Figure 14:

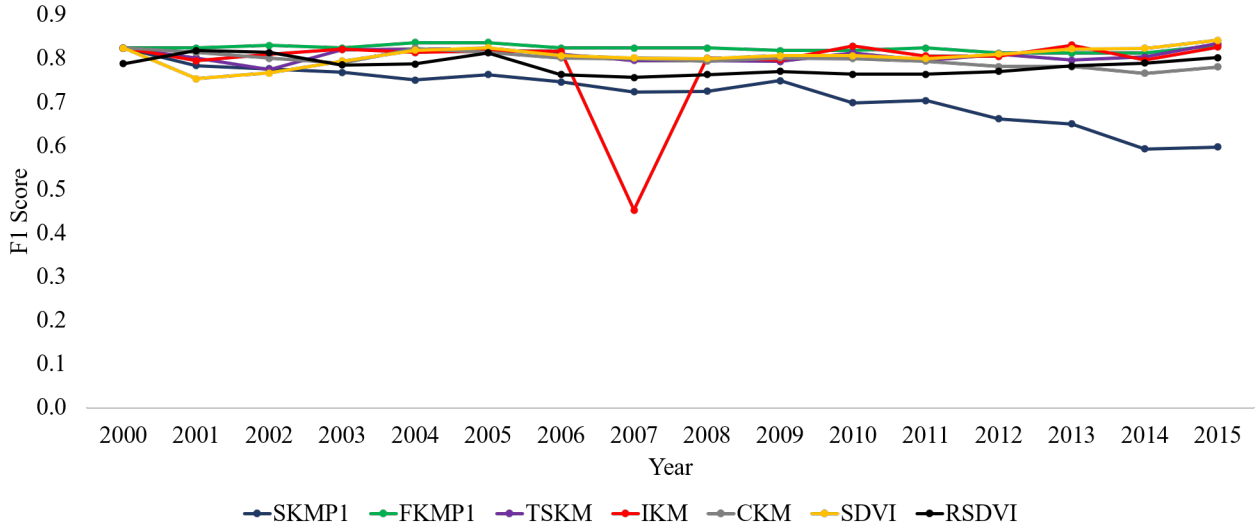

Figure 14: F1-Score performance evolution - Gapminder experiment 1

The previous plot shows some interesting behaviors. First, as shown by the general performance measures, most of the techniques have a very similar behavior, excluding the static segmentation technique. As expected, the static segmentation starts with a good performance, but in later periods its performance is not very good, this is due to the fact that the population is evolving and

therefore it is clear that the segments should not be static.

It is also important to note that, although our non-robust proposed dynamic segmentation technique is not the one with the best performance at the beginning and tends to have slightly worse metrics, during the final periods the performance of our technique is among the best, and in particular in the final period it presents the best F1-Score. This shows that our technique is using past information to improve its results through time, which is the expected behavior.

Finally, we see a clear outlier behavior in 2007 for the IKM technique. This can happen because, in this technique, the centroids are initialized randomly for each period. It is known that the k-means technique is sensitive to initialization such as it is mentioned in Li (2011), therefore what happens during this period is something that can happen when centroids are randomly reinitialized and it can present a serious weakness of this technique.

To proceed, Table 12 shows the run times of each of the techniques. The patterns found in the simulated experiments are also present here: the CKM technique has the highest computation time, followed by RSDVI. We highlight that run times for TSKM and FKMP1 (which were not included in the simulated dataset experiments) are not very high when compared to the rest of the techniques.

Table 12: Median run times of each technique (5 runs) - Experiment 1

Technique	**Execution time (seconds)**
SKMP1	0.677
FKMP1	0.234
TSKM	3.507
IKM	3.210
CKM	38.582
SDVI	5.961
RSDVI	15.076

Finally, in Figure 15, we present the clusters found by our non-robust dynamic segmentation technique. It can be seen that the composition of these clusters is similar to the one of the real groups shown in Figure 13; for example, the technique was able to identify that the middle group (blue group) contains more observations than the other two groups, which is consistent with the reference groups determined by the UNDP based on level of development.

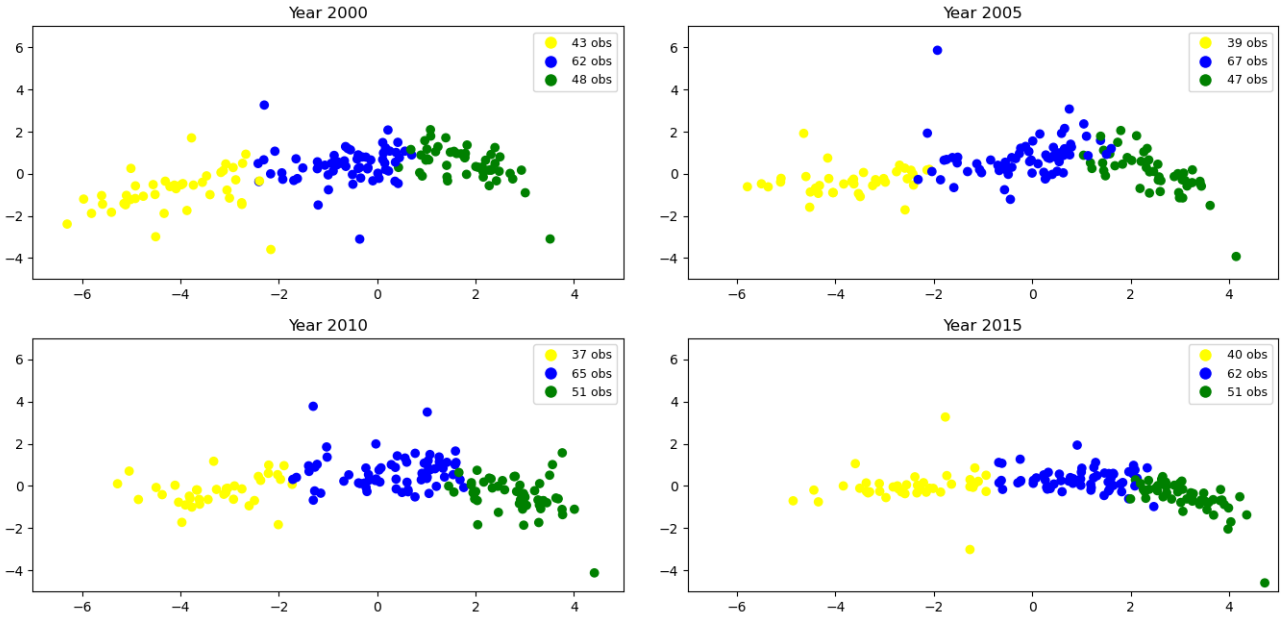

Figure 15: Clusters found with the non-robust proposed technique for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

5.2.4 Experiment 2

For the second experiment, we decided to use the four tiers of human development based on the Human Development Index (HDI) as described by the 2010 Human Development Report published by the UNDP (2010). These tiers establish that each of the quartiles of the HDI represents a country classification.

Countries with a HDI higher than the third quartile have a very high development, countries between the median and the third quartile are considered to have a high development, countries between the first quartile and the median have a medium development and countries placed below the first quartile have a low development. These groups are considered to be the real classification labels for this experiment, and thus these are the real reference labels used to calculate the performance measures for each technique.

In this case, we consider the periods between 1980 and 2015 as our dataset, this is done to analyze how the performance of some techniques changes when many periods are included. Values of different economic and socio-demographic variables for 171 countries for each of these periods are therefore used by the clustering techniques to find the groups for each period, and these groups are then compared with the real grouping described before to calculate the performance of each technique. The variables that we include in the dataset are variables that are related with human development, but we also decided to include in this case some variables related to population (which is not directly tied to human development), this is done to test how well each technique can perform when there are some variables that are not very important. The variables used for this experiment are described in Table 10.

As in the previous experiment, each segmentation technique was applied to the defined dataset for

this experiment. Table 13 contains the performance measures for each of the techniques in this experiment:

Table 13: Overall metrics - Gapminder experiment 2

	SKMP1	FKMP1	TSKM	IKM	CKM	SDVI	RSDVI
Frobenius norm	2064.028	1811.275	1566.003	1411.734	1581.305	1723.889	1388.879
Diagonal Euclidean distance	1588.077	1464.904	1290.435	1157.843	1276.522	1413.817	1062.167
Accuracy	0.513	0.655	0.649	0.670	0.597	0.659	0.692
Precision	0.548	0.518	0.615	0.653	0.602	0.580	0.700
Recall	0.514	0.634	0.635	0.662	0.592	0.641	0.693
F1-Score	0.512	0.563	0.615	0.656	0.595	0.588	0.688

In this case, the robust version of the proposed technique (RSDVI) is the one with the clear best performance, while the non-robust version has a poor performance. This may be because, due to the addition of the variables related with population, there are some countries that may represent outliers due to extreme population (India or China) or very high population density (Singapore), but these countries do not represent outliers in level of development (due to the fact that variables related to population are not used to calculate the HDI). Therefore, in this experiment, it is reasonable that robust techniques perform better than non-robust ones.

Figure 16 displays the real reference groups. As it can be seen, there are some extreme datapoints in the Y axis that may correspond to outliers (in particular, the lower two points correspond to India and China, and the one located on the upper right is Singapore). It is also important to mention that, since not all countries of the world were included, the number of observations in each group is not exactly 25% of the total observations.

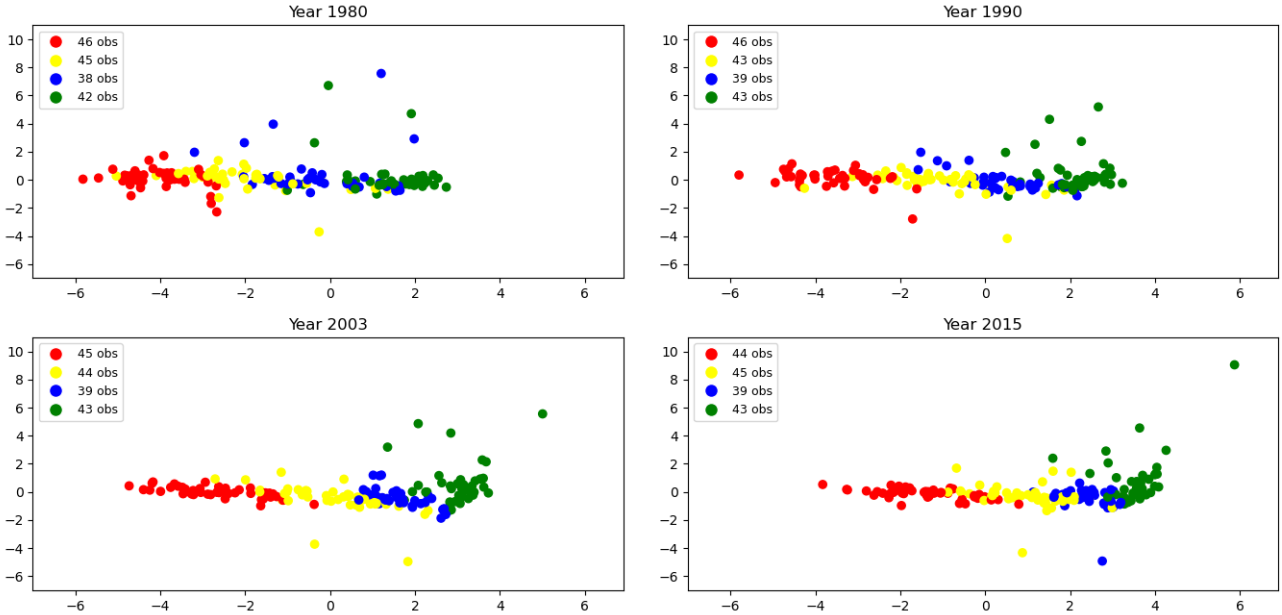

Figure 16: Reference classification groups for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

Table 14 shows the execution times of each of the techniques. Considering that in this experiment

the analyzed period of time is considerable larger than in the previous one (1980-2015 instead of 2000-2015), it is possible to see that run times are much higher than in the previous experiment. This indicates that, for bigger datasets, the execution time of the techniques must be considered as a factor to decide which technique should be used. However, the rank of the techniques does not present significant changes (CKM is still the slowest technique, followed by RSDVI).

Table 14: Median run times of each technique (5 runs) - Experiment 2

Technique	Execution time (seconds)
SKMP1	1.503
FKMP1	0.344
TSKM	10.999
IKM	9.857
CKM	301.111
SDVI	19.305
RSDVI	44.523

Now, Figure 17 shows the evolution of the F1-Score of each of the techniques through the periods:

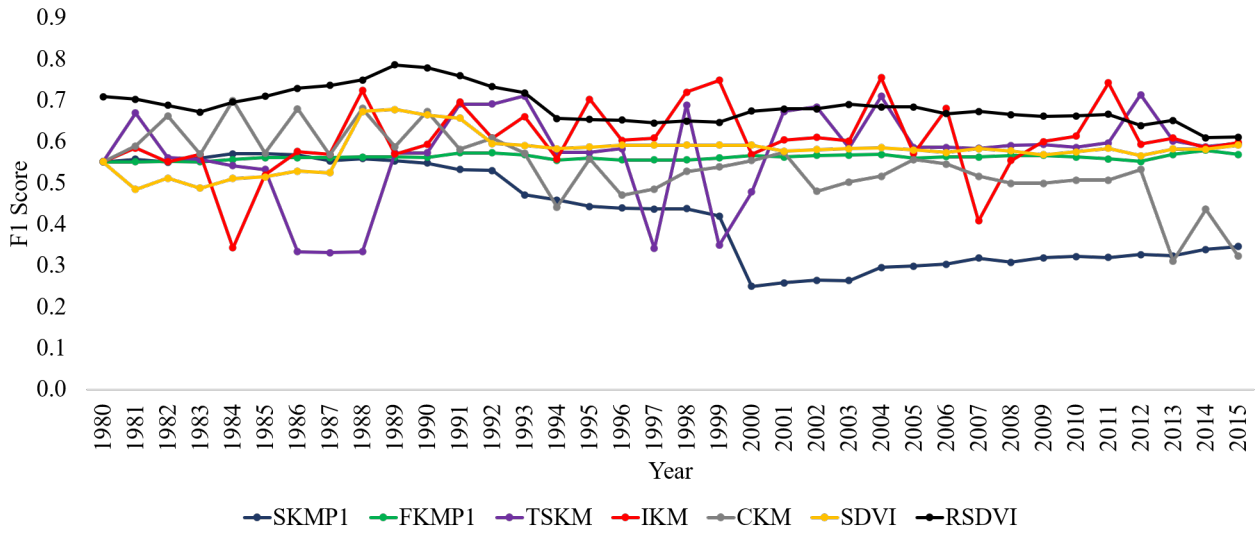

Figure 17: F1-Score performance evolution - Gapminder experiment 2

The previous figure supports the idea that the RSDVI has the best performance, because this is the one which has the highest F1-score in most of the periods, including the last one, added to the fact that in every period it is between the top 3 techniques with best performance. However, it is important to mention that RSDVI results are slightly worse in the final periods, which may indicate that this technique must be readjusted after many periods have passed.

Moreover, it can be seen that IKM and TSKM present decent scores and in a few periods perform better than RSDVI, but they are more unstable because they also have periods with very poor performances. SDVI also exhibits some periods with poor performance at the beginning, which is explained due to its non-robust nature. On the other hand, FKMP1 presents stable results throughout the periods (because the assignments never change), but its performance is not as good

in comparison to the other techniques.

Also, we mention that SKMP1 and CKM present decent results at the beginning but during the final periods they present a very poor performance. These techniques are not able to adapt well to changes through time in this case.

Next, Figure 18 shows the clusters obtained by SDVI for some periods. This image shows the problem that SDVI has in this experiment: for three of the displayed years, SDVI considered the two outliers located on the bottom of the plots (corresponding to India and China) as their own group. These datapoints are outliers and do not form a group by themselves according to our real reference classification based on the HDI. Therefore, this creation of a group formed by only two outliers affects in a great way the performance of the technique.

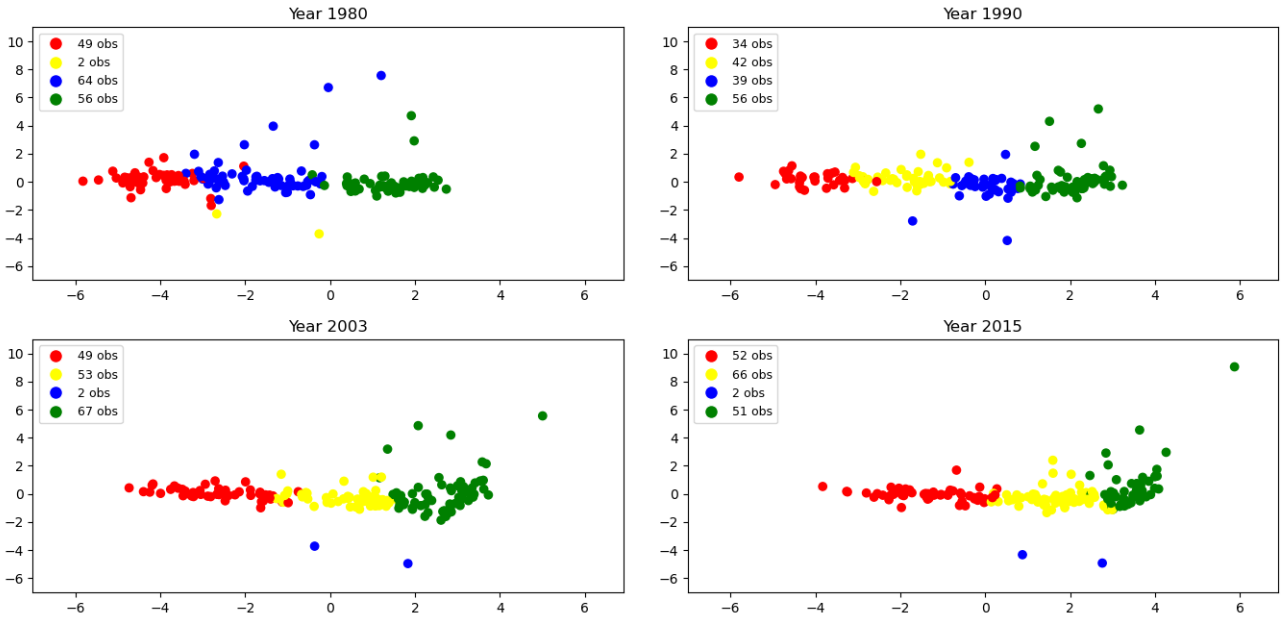

Figure 18: Clusters found with the non-robust proposed technique for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

On the other hand, Figure 19 displays the clusters found by RSDVI. Considering that this technique is robust, the RSDVI is able to determine that, even though the two outlier observations are still present, they should not form a cluster containing just the two of them (the observations are part of other clusters with many more members), therefore minimizing the impact that outliers can have in the segmentation and thus making the technique more stable. It is clear that the clusters found by RSDVI look very similar to the real reference classification shown in Figure 16, confirming the fact that the robust version of proposed the technique is indeed a more stable approach for the problem of dynamic segmentation.

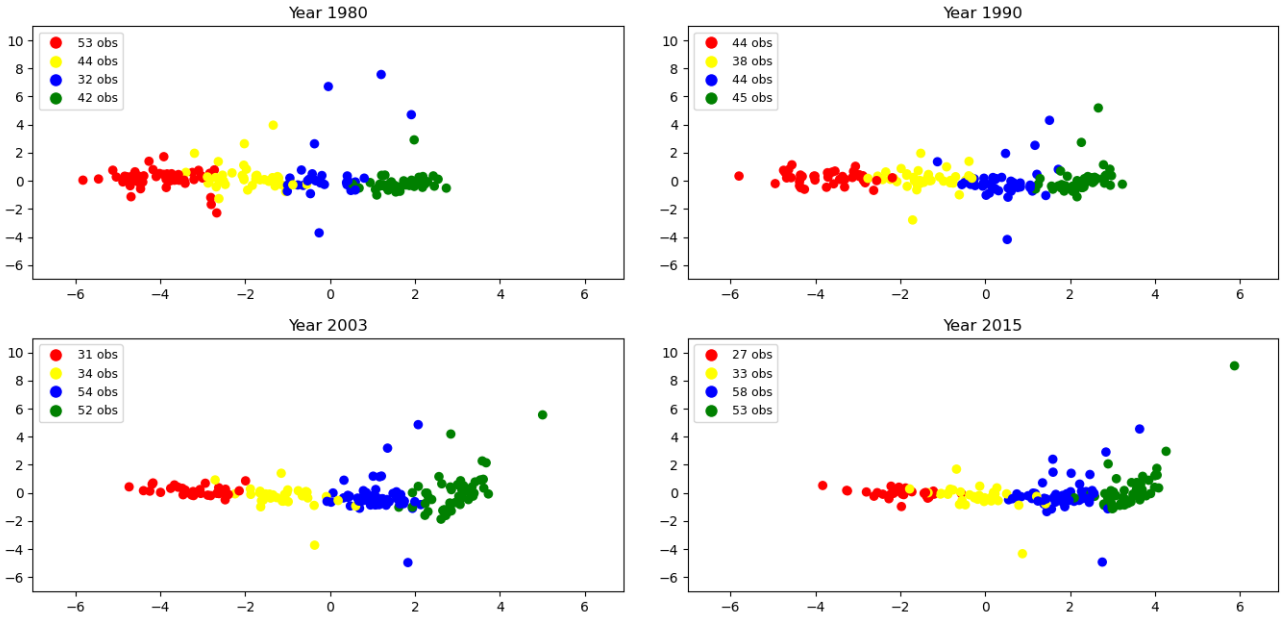

Figure 19: Clusters found with the robust proposed technique for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

5.2.5 Experiment 3

For the third and last experiment, instead of using the HDI for the reference classification, we propose the use of the *World Bank Country classification* (WBCC) by income as the real groups. In this scenario, four groups are considered, in which countries can be classified as: low income, lower-middle income, upper-middle income and high income. This classification can also be related to the level of development of the countries as mentioned in Nielsen (2011).

In order to define which country belongs to each group, we used the WBCC based on the GNI per capita as presented by the World-Bank (2020). The groups per year given by this classification are assigned according to some thresholds defined by the WBCC on the GNI per capita of each country. It is important to mention that, every year, the thresholds used for the segments are updated in the WBCC in order to do adjustments due to inflation, which is what makes these segments dynamic.

The WBCC gathered for this study contains the classification of countries for the periods between 1987 and 2019. It is important to mention that there is not a classification for every country for each year (since there are some classifications missing for some of the countries). In addition, the Gapminder data that is used has as well missing economic and socio-demographic data for some of the countries, which leads to these countries being discarded. Therefore, this experiment is held between 2000 and 2016 and includes observations corresponding to 113 countries. For the implementation of the segmentation techniques, economic and demographic variables that can characterize the countries and their development level are included. The variables included for this experiment are shown in Table 10.

As usual, each segmentation technique was applied to the defined dataset for this experiment. Table 15 shows the performance measures calculated on the global confusion matrix for each technique.

Table 15: Overall metrics - Gapminder experiment 3

	SKMP1	FKMP1	TSKM	IKM	CKM	SDVI	RSDVI
Frobenius norm	933.139	721.364	537.706	566.863	456.287	467.014	493.269
Diagonal Euclidean distance	687.136	523.498	423.175	443.285	366.467	374.152	373.607
Accuracy	0.386	0.514	0.585	0.581	0.636	0.632	0.667
Precision	0.526	0.585	0.620	0.624	0.638	0.630	0.660
Recall	0.416	0.532	0.580	0.578	0.618	0.619	0.653
F1-Score	0.345	0.518	0.575	0.567	0.619	0.615	0.640

In the previous table, it can be seen that the three best techniques correspond to RSDVI, CKM and SDVI. Considering the F1-Score as the principal performance measure, it is possible to see that the robust version of the proposed technique has a better performance overall, followed by CKM and then SDVI. Also, one more time the SKMP1 technique is outperformed by all other approaches, an expected result since it is a static segmentation technique. In addition, the IKM, TSKM and FKMP1 techniques also exhibit a good behavior in comparison to the static techniques, but do not outperform the proposed technique on its non-robust and robust version.

Figure 20 displays the real reference groups for this experiment. It is possible to visualize the four groups: the red group corresponds to countries classified as lower income, the yellow group corresponds to lower-middle income, the blue group reflects the upper-middle income group and the green one corresponds to the high income group.

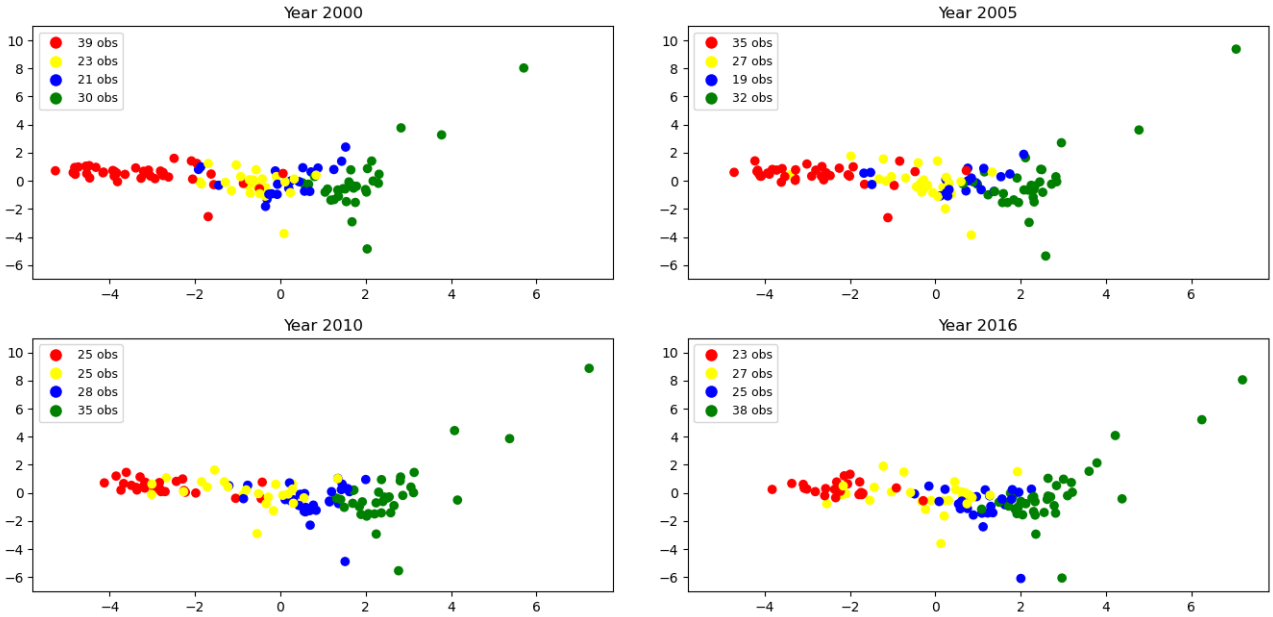

Figure 20: Reference classification groups for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

For a better understanding of the previous figure, it is important to keep in mind that the first two principal components are plotted, which explains why it is possible to visualize some groups whose observations tend to be in the middle of other groups, in particular for the yellow and blue clusters. In addition, it must also be considered that these reference groups are created looking directly at economic factors, but also other variables related to development were included in the dataset used

for this experiment (these other variables probably affect the principal components and therefore the visualization, but they do not have a direct effect on the real reference classification).

In addition to the overall metrics, in Figure 21 it is possible to see the behavior of the F1-score on each of the years in consideration.

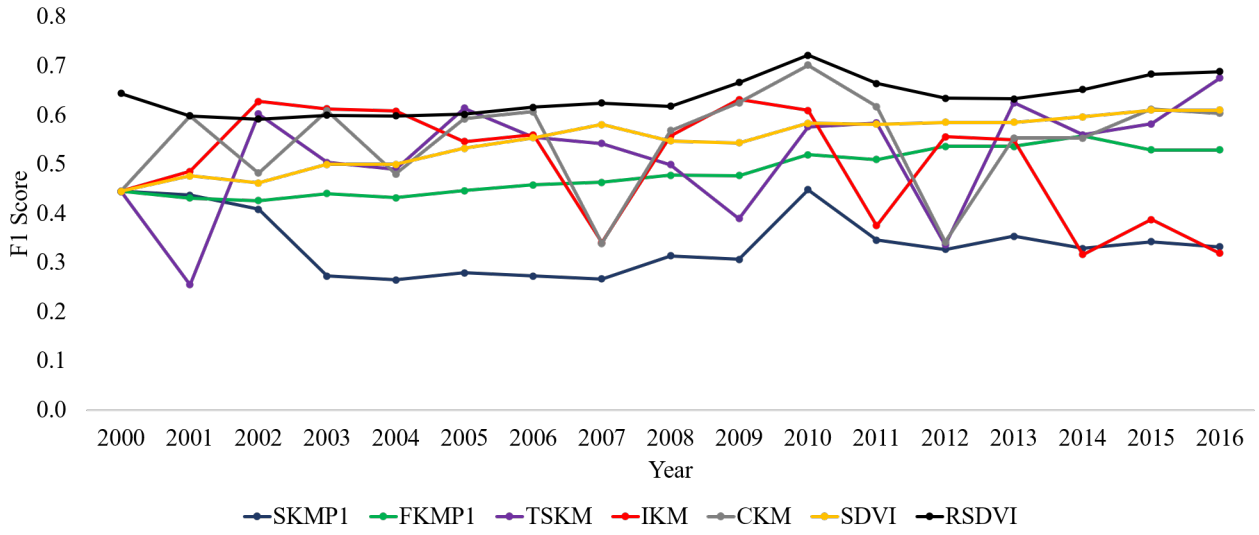

Figure 21: F1-Score performance evolution - Gapminder experiment 3

In the previous Figure, it is clear that the robust version of the proposed technique has the best behavior on the F1-score across almost all periods. Also, it is possible to see that the robust and non-robust version of the proposed technique are stable on their performance during the period under study (and even improve their performance over time), which represents a good behavior.

Next, Table 16 shows the median run times for each technique in this experiment. It is possible to see one more time that the technique that takes longer on its execution is the CKM, followed by the proposed technique in its robust version. Although the proposed technique in its robust and non-robust version are between the slowest techniques, they just require a few seconds for this experiment, which suggests that they can be used without problem for datasets of this size.

Table 16: Median run times of each technique (5 runs) - Experiment 3

Technique	Execution time (seconds)
SKMP1	0.558
FKMP1	0.187
TSKM	3.235
IKM	2.837
CKM	33.233
SDVI	5.324
RSDVI	13.224

For a visual representation of the results of this scenario, Figures 22 and 23 show the plot of the

segmentation given by SKMP1 and RSDVI for some of the years of the dataset.

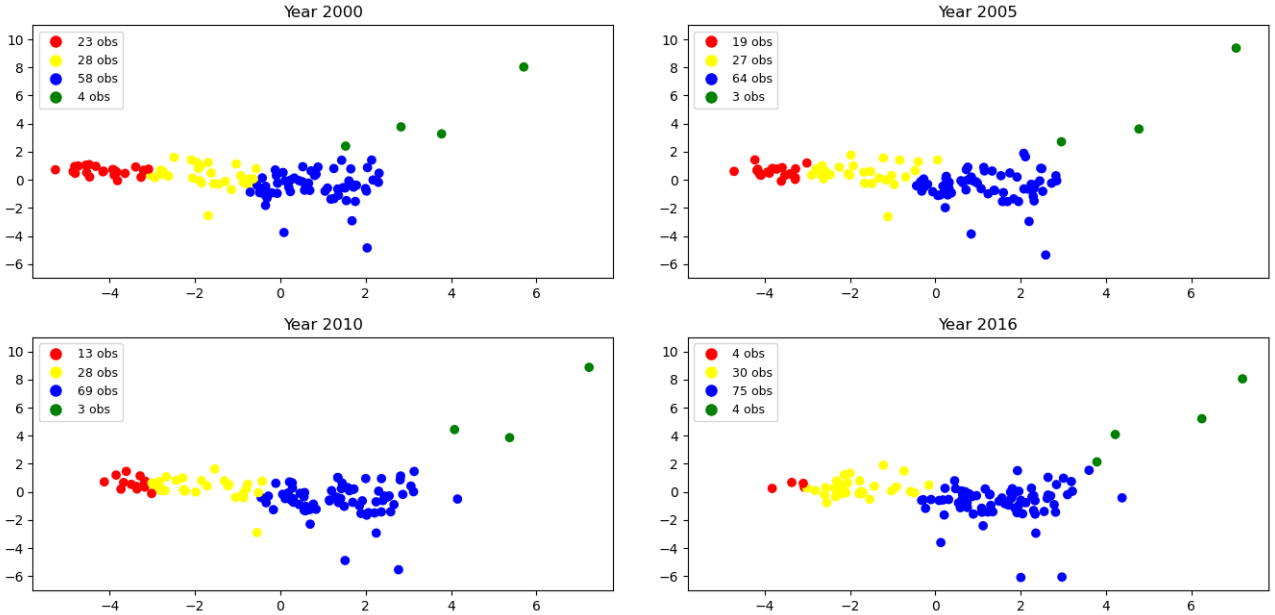

Figure 22: Clusters found with SKMP1 for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

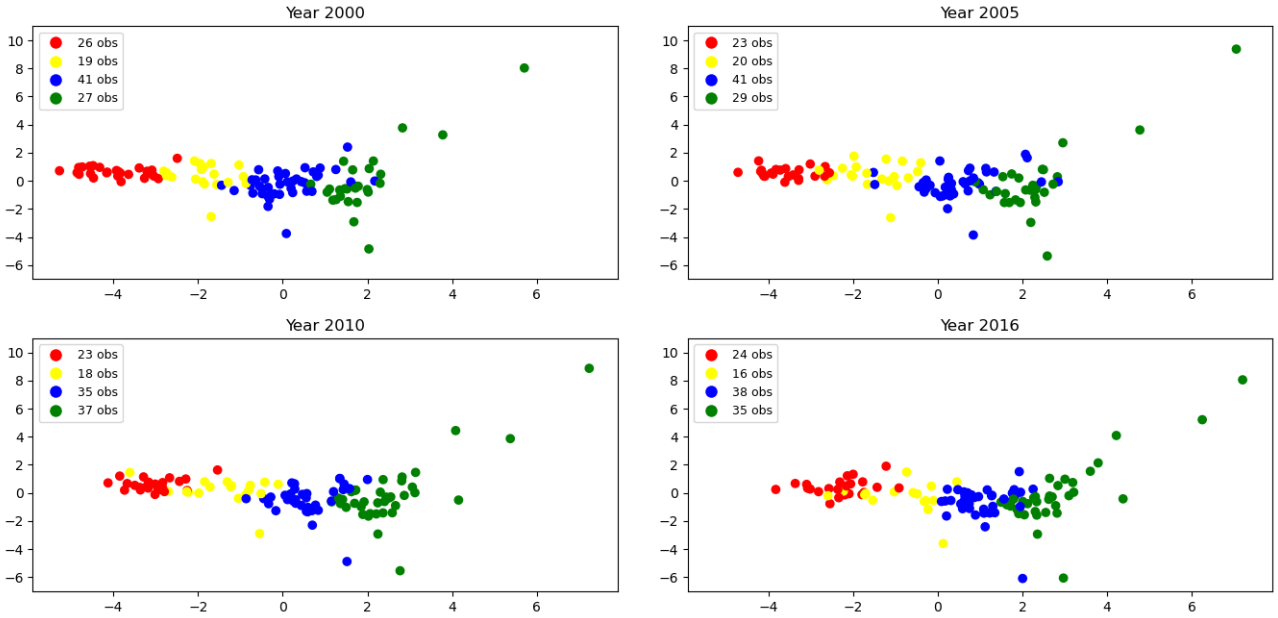

Figure 23: Clusters found with RSDVI for some of the years of this experiment. The first two principal components are plotted. The legend shows the number of observations in each group

From the previous figures, it is possible first to see that SKMP1 considers some outliers as a group (green group), whereas the RSDVI does not make this mistake, which allows the RSDVI to find clusters that are more similar to the real ones shown in Figure 20.

In addition, the SKMP1 does not detect well the dynamism on the segmentation, since it tends to move countries with lower income to groups of countries with a higher income as time passes. This occurs because each year, due to inflation and market fluctuation, the monetary variables of a country tend to increase its value, whereas the technique keeps the same centroids from the first period and never updates them, therefore not adapting through time.

These movements are seen in Figure 22, where many of the observations of the red group tend to move to the yellow group, and the ones from the yellow group tend to move to the blue group, which leads to a final grouping in which there are too many observations in the blue group and too few in the red group. This is not the expected behavior, and this is why each year the GNI per capita thresholds in the WBCC are updated in order to prevent this from happening, which is something that the SKMP1 does not do.

On the other hand, the RSDVI can adapt to the dynamism in the segmentation since it is able to adjust the variables of each cluster to reflect the global evolution of the population through time. This is consistent with the real reference groups, in which the thresholds used to determine these groups are updated every year to account for inflation, and the RSDVI mimics this by updating the segments after each period. Therefore, the proposed technique is able to capture the dynamism of the segmentation, and thus is able find groups that are similar to the real reference classification.

6 Conclusions and Future work

As shown in this study, there exists a gap in the existent dynamic segmentation techniques in the literature, which occurs because they either consider each past period as being as important as the current period or they ignore the past information, which are not the most adequate behaviors. The new proposed segmentation approach using dynamic variables on individuals (SDVI) fills this gap by focusing on balancing the importance of both past information (by weighting the variables based on their importance in previous periods) and present information (by clustering the weighted real observed values of each datapoint in each period). This allows it to outperform other techniques of both static and dynamic segmentation found in the literature in most of the experiments, both for simulated and real datasets.

It was also clear that the modifications made to the segmentation approach using dynamic variables on individuals to make it more robust (RSDVI) are effective because they lead to a better and more stable performance when case-wise or cell-wise outliers are present in the dataset. Moreover, when looking at the evolution of its performance measures and the clusters found by the technique during the periods under study in the real dataset experiments, it was clear that the RSDVI is able to present results that are more stable than both its non-robust counterpart and other dynamic segmentation techniques existent in the literature.

On the other hand, the robust technique behavior was not analyzed when both cell-wise and case-wise outliers occur at the same time, which is something interesting that could be explored in future researches. Execution time is another factor that should be checked because it is more than

two times higher than its non-robust counterpart. Also, we consider that future research can also be directed towards demonstrating if the proposed robust approach satisfies some of the important mathematical properties that a robust technique should have or if a different approach is needed.

Finally, we consider that future studies should be conducted on the proposed technique on some of these topics: a sensitivity analysis of its parts, the use of the technique in big data applications, a creation of a general framework to solve dynamic segmentation problems using this technique or even modifying this approach to be able to predict the composition of clusters in future periods.

References

Alqallaf, Fatemah, Van Aelst, Stefan, Yohai, Victor J, Zamar, Ruben H, *et al.* 2009. Propagation of outliers in multivariate data. *The Annals of Statistics*, **37**(1), 311–331.

Angelin, B, & Geetha, A. 2020. Outlier Detection using Clustering Techniques–K-means and K-median. *Pages 373–378 of: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE.

Benítez, Ignacio, Quijano, Alfredo, Díez, José-Luis, & Delgado, Ignacio. 2014. Dynamic clustering segmentation applied to load profiles of energy consumption from Spanish customers. *International Journal of Electrical Power & Energy Systems*, **55**, 437–448.

Blocker, Christopher P., & Flint, Daniel J. 2007. Customer segments as moving targets: Integrating customer value dynamism into segment instability logic. *Industrial Marketing Management*, **36**(6), 810 – 822.

Cheng, Ching-Hsue, & Chen, You-Shyang. 2009. Classifying the segmentation of customer value via RFM model and RS theory. *Expert systems with applications*, **36**(3), 4176–4184.

Crespo, Fernando, & Weber, Richard. 2005. A methodology for dynamic data mining based on fuzzy clustering. *Fuzzy Sets and Systems*, **150**(2), 267 – 284.

Gapminder. 2020. *Gapminder*. `https://www.gapminder.org/data/`. Online; October 15, 2020.

Haley, Russell I. 1968. Benefit Segmentation: A Decision-Oriented Research Tool. *Journal of Marketing*, **32**(3), 30–35.

Helsen, K., & Green, P. 1991. A Computational Study of Replicated Clustering with an Application to Market Segmentation. *Decision Sciences*, **22**, 1124–1141.

Huang, Jih-Jeng, Tzeng, Gwo-Hshiung, & Ong, Chorng-Shyong. 2007. Marketing segmentation using support vector clustering. *Expert Systems with Applications*, **32**(2), 313 – 317.

Kuo, R.J., Ho, L.M., & Hu, C.M. 2002. Integration of self-organizing feature map and K-means algorithm for market segmentation. *Computers Operations Research*, **29**(11), 1475 – 1493.

Li, Chun. 2011. Cluster Center Initialization Method for K-means Algorithm Over Data Sets with Two Clusters. *Procedia Engineering*, **24**(12), 324–328.

Munusamy, Sivaguru, & Murugesan, Punniyamoorthy. 2020. Modified dynamic fuzzy c-means clustering algorithm–Application in dynamic customer segmentation. *Applied Intelligence*, 1–21.

Nielsen, Lynge. 2011. *Classifications of Countries Basedon their Level of Development : How it is Done and How it Could Be Done.*

Pereira, Gonçalo, & Mendes-Moreira, João. 2016. Monitoring clusters in the telecom industry. *Pages 631–640 of: New Advances in Information Systems and Technologies.* Springer.

Punj, Girish. 1983. Cluster Analysis in Marketing Research: Review and Suggestions for Application. *Journal of Marketing Research*, **20**(05).

Rezaee, Mustafa Jahangoshai, Jozmaleki, Mehrdad, & Valipour, Mahsa. 2018. Integrating dynamic fuzzy C-means, data envelopment analysis and artificial neural network to online prediction performance of companies in stock exchange. *Physica A: Statistical Mechanics and its Applications*, **489**, 78 – 93.

Rousseeuw, Peter. 1984. Least Median of Squares Regression. *Journal of The American Statistical Association - J AMER STATIST ASSN*, **79**(12), 871–880.

Sasaki, Yutaka, *et al.* 2007. The truth of the F-measure. *Teach Tutor mater*, **1**(5), 1–5.

Sharma, Arun, & Lambert, Douglas. 1994. Segmentation of Markets Based on Customer Service. *International Journal of Physical Distribution Logistics Management*, **24**(01), 50–58.

Smith, Wendell R. 1956. Product differentiation and market segmentation as alternative marketing strategies. *Journal of marketing*, **21**(1), 3–8.

UNDP. 2010. *Human Development Report 2010: The Real Wealth of Nations - Pathways to Human Development. New York.* `http://hdr.undp.org/en/content/human-development-report-2010"`.

UNDP. 2015. *HDI Series Cartagena.* `http://hdr.undp.org/sites/default/files/hdi_series_cartagena.xlsx`. Online; January 11, 2021.

UNDP. 2018. *Triennial review dataset 2000 - 2018.* `https://www.un.org/development/desa/dpad/wp-content/uploads/sites/45/page/LDC_data.xls`. Online; January 11, 2021.

UNDP. 2020. *Human Development Index (HDI).* `http://hdr.undp.org/en/content/human-development-index-hdi`. Online; January 11, 2021.

Velasco, Henry, Laniado, Henry, Toro, Mauricio, Leiva, Víctor, & Lio, Yuhlong. 2020. Robust three-step regression based on comedian and its performance in cell-wise and case-wise outliers. *Mathematics*, **8**(8), 1259.

Vollering, Jan B. 1984. Interaction based market segmentation. *Industrial Marketing Management*, **13**(2), 65 – 70.

World-Bank. 2020. *World Bank GNI per capita Operational Guidelines Analytical Classifications.* `https://databank.worldbank.org/data/download/site-content/OGHIST.xls"`.

Zhang, Rui, Jin, Zhigang, Xu, Peixuan, & Liu, Xiaohui. 2019. A dynamic clustering based method in community detection. *Cluster Computing*, 1–15.