Revista de Administração Contemporânea
Journal of Contemporary Administration

ASSOCIAÇÃO NACIONAL DE PÓS-GRADUAÇÃO E PESQUISA EM ADMINISTRAÇÃO

e-ISSN: 1982-7849

Tutorial Article

Cluster Analysis in Practice: Dealing with Outliers in Managerial Research

Análise de Clusters na Prática: Lidando com *Outliers* na Pesquisa Gerencial

Humberto Elias Garcia Lopes[1]
Marlusa de Sevilha Gosling[2]

ABSTRACT

Context: in recent years, cluster analysis has stimulated researchers to explore new ways to understand data behavior. The computational ease of this method and its ability to generate consistent outputs, even in small datasets, explain that to some extent. However, researchers are often mistaken in holding that clustering is a terrain in which anything goes. The literature shows the opposite: they must be careful, especially regarding the effect of outliers on cluster formation. **Objective:** in this tutorial paper, we contribute to this discussion by presenting four clustering techniques and their respective advantages and disadvantages in the treatment of outliers. **Methods:** for that, we worked from a managerial dataset and analyzed it using *k-means*, PAM, DBSCAN, and FCM techniques. **Results:** our analyzes indicate that researchers have distinct clustering techniques for dealing with outliers accordingly. **Conclusion:** we concluded that researchers need to have a more diversified repertoire of clustering techniques. After all, this would give them two relevant empirical alternatives: choose the most appropriate technique for their research objectives or adopt a multi-method approach.

Keywords: cluster analysis; outliers; *k-means*; DBSCAN; fuzzy clustering.

RESUMO

Contexto: nos últimos anos, a análise de clusters tem estimulado os pesquisadores a explorar novas maneiras para entender o comportamento dos dados. A facilidade computacional desse método e sua habilidade de gerar resultados consistentes, mesmo em bases de dados pequenas, explicam isso em certa medida. Entretanto, os pesquisadores frequentemente se equivocam ao sustentar que a clusterização é um território no qual vale tudo. A literatura mostra o oposto: eles têm que ser cuidadosos, especialmente em relação ao efeito dos *outliers* na formação dos clusters. **Objetivo:** neste artigo tutorial, nós contribuímos para essa discussão ao apresentarmos quatro técnicas de clusterização com suas respectivas vantagens e desvantagens no tratamento dos *outliers*. **Métodos:** para isso, nós trabalhamos com uma base de dados gerenciais, analisando-a por meio das técnicas *k-means*, PAM, DBSCAN e FCM. **Resultados:** nossas análises indicam que os pesquisadores têm diferentes técnicas de clusterização ao seu dispor para tratar os *outliers* adequadamente. **Conclusão:** nós concluímos que os pesquisadores precisam ter um repertório mais diversificado de técnicas de clusterização. Afinal, isso daria a eles duas alternativas empíricas relevantes: escolher a técnica mais apropriada para os objetivos das suas pesquisas ou adotar uma abordagem multimétodo.

Palavras-chave: análise de clusters; *outliers*; *k-means*; DBSCAN; clusterização difusa.

1. Pontifícia Universidade Católica de Minas Gerais, Programa de Pós-Graduação em Administração, Belo Horizonte, MG, Brazil.
2. Universidade Federal de Minas Gerais, Faculdade de Ciências Econômicas, Belo Horizonte, MG, Brazil.

Cite as: Lopes, H. E. G., & Gosling, M. S. (2021). Cluster analysis in practice: Dealing with outliers in managerial research. *Revista de Administração Contemporânea*, 25(1), e200081. https://doi.org/10.1590/1982-7849rac2021200086

JEL Code: A1, C1, M1.

Editor-in-chief: Wesley Mendes-Da-Silva (Fundação Getulio Vargas, EAESP, Brazil)
Associate Editor: Henrique Castro Martins (PUC Rio, IAG, Brazil)
Reviewers: Pablo Cristini Guedes (Universidade Federal do Rio Grande do Sul, Brazil)
One of the reviewers chose not to disclose his/her identity.

Received: March 27, 2020
Last version received: August 19, 2020
Accepted: August 21, 2020

# of invited reviewers until the decision:

	1	2	3	4	5	6	7	8	9	10
1st round	X	X	X	X	X	✓	X	X	X	✓
2nd round	✓									
3rd round	✓									
4th round	✓									
5th round	✓									

INTRODUCTION

Cluster analysis is one of the most widely known multivariate methods to understand data behavior, including in the managerial area (Aggarwal, 2014; Ketchen & Shook, 1996). This popularity is due to the intuitive concepts that underpin the method and facilitate the interpretation of outputs. In managerial research, the possibility of applying the method exceeds solving specific problems such as segmenting markets or identifying different patterns of behavior among the subjects of research. Cluster analysis also attracts researchers in the area because it does not demand large datasets or data that meet the more restrictive assumptions of other multivariate methods, such as linearity, normality, and homoscedasticity (Norusis, 2006a).

Nevertheless, this does not mean that anything goes in cluster analysis: in fact, specific patterns of data behavior can bias the outputs substantially. One of these patterns results from the presence of outliers, defined as abnormal values that can have an extreme effect on the analysis (Acock, 2014; Irizarry & Love, 2015). It is not always easy to identify these outliers correctly, nor to estimate their real influence on data behavior (Adams, Hayunga, Mansi, Reeb, & Verardi, 2019; Loperfido, 2020). For this reason, researchers often follow protocols that recommend excluding these outliers, treating them as a problem to be solved before starting the core of the analysis (Hair, Black, Babin, & Anderson, 2018; Malhotra, 2018).

However, these protocols are not the only option for researchers. Indeed, statisticians and mathematicians continue to develop clustering techniques that treat outliers without necessarily excluding them from the analysis. That is a methodological advance, since these outliers are information and, as such, can help the researcher to understand data behavior. Thus, not always the most recommended path will be their mere exclusion from the analysis.

The purpose of this paper is to help researchers learn about four clustering alternatives. By doing that, we can provide them the knowledge to deal with outliers as part of the research outcome and not as a problem in itself. To do this, we analyze the data from this research using four different techniques: *k-means*, PAM, DBSCAN, and FCM.

In this paper, we had two concerns. The first was using actual data from managerial research. Ordinarily, tutorial papers on multivariate methods use datasets created from the random selection of a continuous dataset and which often obey a particular distribution, such as Gaussian. On the one hand, this strategy makes the technique work perfectly, generating exemplary outputs. On the other hand, it moves away from the reality of many researchers, often involved with data that are far from meeting the behavior prescribed in data science manuals. That is the case with managerial research, which regularly uses discrete data and does not meet the distributional demands of many multivariate methods. To get closer to this reality, we use the original dataset of Lopes, Pereira and Vieira (2009).

Our second concern was to be instructive. Tutorial papers on quantitative methods and techniques often lose their way in explanations with much-advanced mathematics that confuse rather than clarify the subject. To avoid this, we wrote a more fluid and friendly text, which was intelligible even to researchers less familiar with clustering. That has led us to limit the mathematical discussion to what was strictly necessary. We also focused on showing the application of our results to management analysis, avoiding tiring the reader with overly technical explanations for a tutorial paper. Finally, we ensured the reproducibility of our outputs by making our dataset and analysis codes available to readers as supplemental material. These codes are in R language, widely used in academic circles.

LITERATURE REVIEW

Cluster analysis is a multivariate exploratory method widely used in several areas since the 1960s (Scoltock, 1982). It classifies objects, allocating them into internally homogeneous but also heterogeneous groups (Everitt & Hothorn, 2006). Therefore, its logic is to gather what is similar while separating what is different.

This definition shows that there is no cluster analysis without objects. They are elements that take different forms, such as countries, social groups, individuals, products, or any other element that can be classified from a certain number of attributes (Yu, Wang, Wang, & Zeng, 2020). Since the method is multivariate, it computes these attributes simultaneously. Cluster analysis describes the behavior of objects in groups in the exploratory phase of their investigation since the output is unique to the objects included in the analysis. Consequently, the inclusion or exclusion of any of them from the original research will imply a different output, making the researcher have to resort to other multivariate methods if he or she wants his analysis to be predictive (Fávero & Belfiore, 2017).

Despite this limitation, cluster analysis has been used frequently in many areas. For example, Ketchen and Shook (1996) reviewed the theory and applications of the method in strategic management research. Sun, Chen, Xiong and Guo (2017) concluded that clustering techniques could help researchers identify dynamic decision patterns in organizations and firms. Finally, Thrun (2019) studied the gross domestic product of 160 countries and concluded that an economic event that occurred in 2001 was instrumental in allocating these objects to different groups.

To some extent, this popularity of cluster analysis is due to its less restrictive assumptions compared to those of other multivariate methods (Everitt & Hothorn, 2006). Consequently, two aspects are especially attractive to management researchers: many cluster analysis techniques do not require datasets with many observations or that these data are associated with a specific distribution (Norusis, 2006a). That solves two problems.

To begin with, researchers hardly use large datasets, primarily when they collect data from primary sources. Thus, there is much research with few observations, which restricts the use of multivariate methods such as generalized linear models (GLS) or covariance-based structural equation modeling (COV-SEM), among others. Another aspect is that the data rarely follow a specific distribution. In this respect, quantitative managerial research often uses qualitative variables, i.e., discrete ones. Although they can be treated as quantitative in specific cases (Moustaki, Jöreskog, & Mavridis, 2004; Nunnally & Bernstein, 1994), data hardly meet the distributional requirements of specific multivariate methods. When the researcher insists on using them, he or she assumes the risk of having strongly biased outputs that will have little or no theoretical or practical utility (Husson, Lê, & Pagès, 2017).

The ability of some cluster analysis techniques to generate consistent outputs in small datasets that do not follow distributional assumptions may lead the researcher to believe that 'anything goes' in this method, which is a misconception. It has two aspects that should not be ignored: the standardization of the scales of variables and, prominently, the effect of outliers.

For the former, cluster analysis forms the groups from the data of the quantitative variables, which must be in the same measurement unit. It is adequate to have qualitative variables, but they must be identifiers of the groups and not part of the calculations that allocate objects to them (Maechler, 2019). For example, the nominal qualitative variable 'firm' will help the researcher know that firms *X* and *Y* belong to Cluster 1 and that firms *K* and *Z* are in Cluster 2. However, the data regarding their names were not used to form these groupings because they are not numerical, being outside the calculations. At the same time, all quantitative variables must be in the same measurement unit. Otherwise, the clustering algorithm will use different magnitudes and may bias the results. Fortunately for researchers, the procedure to avoid this problem is simple: to standardize the quantitative variables, granting them values on the same scale.

The most prominent challenge of cluster analysis lies in the second aspect: assessing the effect of outliers on the output. In the computational logic of this method, the objects allocated in the same cluster need to be spatially close to each other, ensuring internal homogeneity. At the same time, each cluster must diverge from their counterparts; that is, they have to be heterogeneous. However, a dataset with many outliers may compromise both the internal homogeneity and the external heterogeneity of clusters, producing substantially skewed outputs. This risk exists because the outliers are values that escape the usual limits of data variation (Acock, 2014; Irizarry & Love, 2015), impacting to a greater or lesser degree the calculation of the distances that will serve to classify the objects in the clusters. Consequently, if these outliers have a powerful effect, the researcher will not be able to state that the clusters are distinct from each other or that their objects are internally homogeneous (Kassambara, 2017).

The effect of outliers represents an additional challenge for the researcher: the geometrical format of clusters. Cluster analysis covers distinct clustering techniques to deal with specific patterns of data behavior (Ester, Kriegel, Sander, & Xu, 1996). Many of the best-known techniques of management researchers are suitable for dealing with spherical-shaped or convex clusters. That means that specific techniques identify clusters only when they follow these geometric patterns. Otherwise, the output may be ambiguous or even contradictory.

In practice, research data do not always fit into this situation. The researcher may have to analyze data that form groups with intersections, that is, that are not entirely distinct from each other. More than that, there may be cases in which there is more than one geometric format for the groups, which makes it impossible to use the cited techniques. This issue of the impact of outliers on geometric format shows that the researcher should know a broader repertoire of clustering techniques. After all, the volume of data available to researchers has grown exponentially, making their analysis more complicated than in the past (Caffo, 2016).

These challenges motivated us to develop this paper, in which we describe the procedures for the reader to use four clustering techniques that deal with outliers differently. With this, we want to show that this kind of data can be seen as part of the solution, and not as a problem that needs to be solved.

METHODOLOGY

We compared the four clustering techniques from the dataset *Consumer* (Lopes, Pereira, & Vieira, 2009). It contains data on the satisfaction of 2,145 consumers with the services provided by a sample of Brazilian companies, covering indicators associated with perceived quality, perceived value, expectations, complaints, loyalty, and image. We chose to use this dataset so that we could show the practical application of cluster analysis in the managerial area. That would not have been possible if we had used the traditional practice of tutorial papers to create datasets from the random selection of values on a continuous scale.

In this paper, we extracted a predefined test sample with n = 108 consumers — about 5% of the dataset — to have visually accessible plots to analyze. After all, using the 2,145 consumers from the original dataset would produce plots with many overlapping images and therefore confusing for the reader. We also decided to exclude the missing data from this test sample because this subject and its implications would extrapolate the scope of this paper.

The following section describes the procedures we adopted in each clustering technique. We have written the codes in R language, making them available for download along with the original dataset as supplemental material.

DATA ANALYSIS

The following subsections describe and discuss our steps for computing the four clustering techniques covered in this paper. To facilitate the reader's understanding, we briefly comment on each chunk of the code we developed in R. The beginning of the code line is indicated in this text by the symbol R>.

The reader must be aware that this code is supposed to run in the R Studio integrated development environment (IDE). It provides predefined functions that allow the researcher to execute actions in the R language efficiently. One of them is crucial for running the code appropriately: to download and import the dataset. To do that in R Studio, after downloading the dataset to a chosen directory on his or her computer, the researcher has to go to the upper menu and follow this sequence of actions: File/Import Dataset/From Excel.

After that, it is necessary to install the packages for the analysis by running the command:

```
R> install.packages(c("stats", "factoextra",
"fpc","dbscan", "cluster"))
```

The next step is to load those packages with the commands:

```
R> library(stats)
R> library(factoextra)
R> library(fpc)
R> library(dbscan)
R> library(cluster)
```

Those procedures will guarantee that the code runs correctly. We also included extra guidance on the code to help researchers to understand each decision we took on the analysis.

Creating the test sample

The first step in creating the test sample was to omit the missing data from the original base and arm the result into a new dataset:

```
R> ConsumerNoMissing <- na.omit(Consumer)
```

In this code, *ConsumerNoMissing* is the matrix that stores the variables without missing data, created by using the function *na.omit( )* over the original *Consumer* base. Next, we standardize the values of the quantitative variables, which were between columns 2 and 51 of the *ConsumerNoMissing* dataset. To do so, we used the *scale( )* function, selecting the desired columns through the subscript *[seq( )]*.

This procedure allowed the variables to adopt a common scale, which is a prerequisite for cluster analysis techniques (Beysolow, 2017). The code we used was:

```
R> ConsumerScale <- scale(ConsumerNoMissing[seq(2,
51)])
```

Subsequently, we extracted the predefined sample using the *head( )* function, taking the first 108 lines as selection criteria and storing the data in the *ConsumerScaleSample* dataset. That was a crucial step for allowing the reproducibility of results (Peng, 2019).

```
R> ConsumerScaleSample <- head(ConsumerScale, 108)
```

At the end of this procedure, we had a new dataset to start the analysis with the four clustering techniques. We presented the outputs in the following subsections.

Technique 1: *k-means* clustering

The *k-means* is a technique developed in the 1960s that distributes the objects through the partition system in a k number of clusters previously defined by the researcher (MacQueen, 1967). Its basic idea is to minimize the within-cluster variation, that is, the distance between the elements classified in the same cluster, ensuring that it is as homogeneous as possible.

There are a wide number of algorithms that minimize that variation, but Hartigan and Wong (1979) developed the most widely used to date. It defines the within-cluster variation as the sum of squared Euclidian distances between items and their centroid. Mathematically, let x_i be an observation assigned to the cluster C_k and μ_k be the mean value of the observation assigned to the same cluster. Then, the within-cluster variation is:

$$W(C_k) = \sum_{x_i \in C_k} (x_i - \mu_k)^2$$

Therefore, the total within-cluster variation is given by:

$$TW = \sum_{k=1}^{k} W(C_k) = \sum_{k=1}^{k} \sum_{x_i \in C_k} (x_i - \mu_k)^2$$

The *k-means* algorithm estimates the value that minimizes *TW*. For this, the researcher needs to define the initial number of clusters (Sugar & James, 2003). Then, the algorithm will select *k* objects randomly, which will be the centroids. Then, the remaining objects are assigned to their closest centroid, generating a new centroid of each cluster. This process repeats iteratively until obtaining the minimum *TW* value.

The *k-means* have become quite popular among researchers, notably for their relative theoretical simplicity and the ease of interpreting their results (Boehmke & Greenwell, 2019; Janssen & Wan, 2020). However, there is one aspect that deserves attention: the presence of outliers in the dataset may affect the output. That is because each centroid is a mean, that is, a measure of central tendency whose value is affected by extreme values. Thus, researchers must be cautious, even in the face of apparently consistent outputs.

Operationally, our first step to computing *k-means* in R was to fix the optimum number of clusters (*k*). We made this decision from the information of two estimation methods: elbow and average silhouette. The first identifies the number of clusters so that adding more of them implies merely incremental variation on the total within sum of square. That measure represents the clustering solution's compactness that should be as small as possible, assuring the best intra-cluster homogeneity. The second method measures the quality of clustering by determining how well each object lies within its cluster. Hence, the higher the average silhouette, the better the clustering solution (Kassambara, 2017). Additional discussion on the optimal number of clusters is available in the Appendix A.

To compute the elbow method, we used the code with the *fviz_nbclust( )* function applied to the *ConsumerScaleSample* dataset and with the *kmeans* parameters defining the clustering technique and *wss* selecting the elbow method. This code created a plot with the *geom_vline( )* function.

```
R> fviz_nbclust(ConsumerScaleSample, kmeans, method
= "wss") + geom_vline(xintercept = 2, linetype = 2)
+ labs(subtitle = "Elbow method")
```

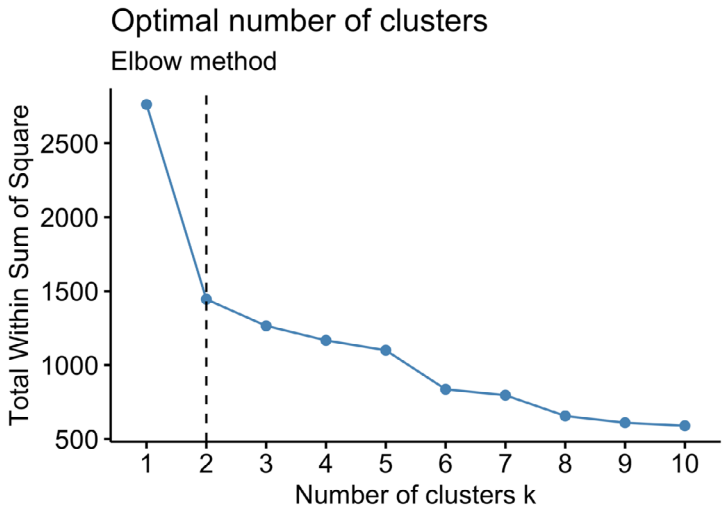

Figure 1. The optimal number of clusters: *k-means* clustering with the elbow method.

In Figure 1, the dotted line indicates the optimal number of clusters. It stands in the point from which the changes in the number of clusters (x-axis) correspond to an incremental variation on the total within sum of square (y-axis). Source: research data.

In Figure 1, the numbers of clusters to the right of the dotted line represent incremental changes in total within sum of square. That means that the researcher will not have a better solution if he chooses any of them as *k*. What matters in the elbow method is the dotted line, as it marks the point of inflection of the curve and, consequently, the optimal number of clusters. For this reason, we considered *k* = 2.

The second method was the average silhouette, computed by the following code and whose output is in Figure 2.

```
R> fviz_nbclust(ConsumerScaleSample, kmeans, method =
"silhouette") + labs(subtitle = "Average silhouette
method")
```

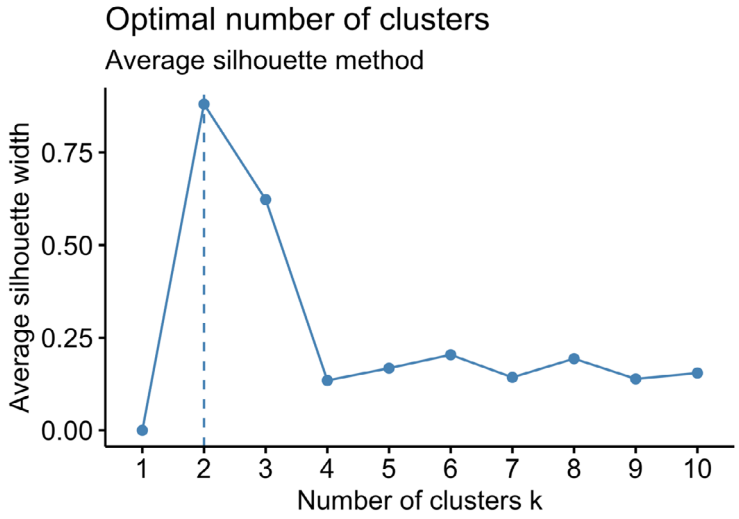

Figure 2. The optimal number of clusters: *k-means* clustering with the average silhouette method.

In Figure 2, the dotted line indicates the optimal number of clusters according to the average silhouette method. That line stands where the number of clusters (x-axis) intercepts the maximum average silhouette width (y-axis). The higher the average silhouette, the better the clustering solution. Source: research data.

In this method, the optimum number of clusters corresponds to the largest average silhouette width, marked with the dotted line. Thus, Figure 2 showed us that we should keep $k = 2$.

We computed the *k-means*, saving the output in *km.Consumer* through the code:

```
R> km.Consumer <- kmeans(ConsumerScaleSample, 2, nstart = 30)
```

It triggered the *kmeans( )* function, in which the digit '2' represented *k* and *nstart* was the number of random starting partitions when centers were a number. It was arbitrary and greater than 1.

To facilitate the analysis, we plotted the results in Figure 3, created with this code:

```
R> fviz_cluster(km.Consumer, data = ConsumerScaleSample,
palette = "jco", ellipse = TRUE, star.plot = TRUE,
repel = TRUE, ggtheme = theme_minimal())
```

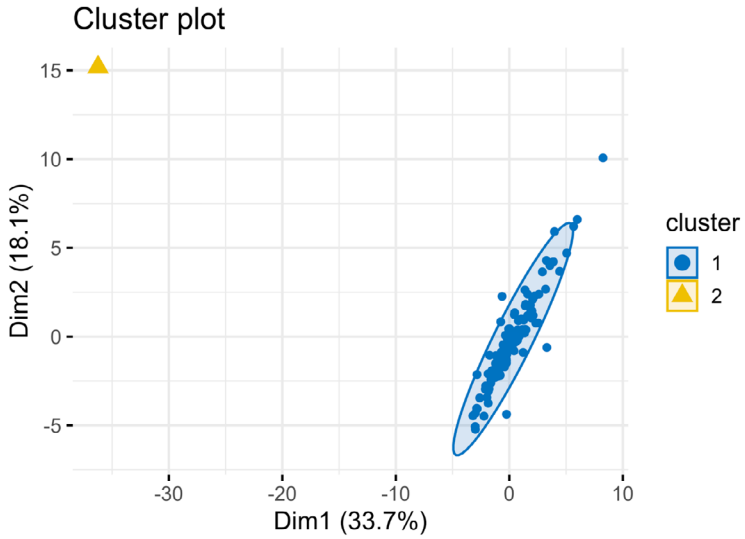

Figure 3. *K-means* clustering with $k = 2$.

Figure 3 shows that the consumers have a significant homogeneous behavior considering the variables in the dataset. They lied in Cluster 1, except for the one single consumer on the high left corner, assigned to Cluster 2. Source: research data.

This output showed that Cluster 1 included almost all consumers, except the one that was comprised within Cluster 2. From a managerial point of view, this output also indicated that most consumers behave very similarly regarding the factors that affect their overall satisfaction with the products. In other words, the firm would be operating in a market with practically a single segment, which would allow it to develop products aimed only at this audience. If it did this, this hypothetical firm could obtain very significant economies of scope and scale (Besanko, Dranove, Shanley, & Schaefer, 2016).

Results with unit clusters or that aggregate almost all objects deserve attention, even if they have essential information for the researcher and are relatively common (Raykov, Boukouvalas, Baig, & Little, 2016). There are several reasons for this extra attention, one of them being the effect of outliers. As the centroid is a mean, its value can be substantially affected by extreme values in the dataset, causing the *k-means* clustering algorithm to identify clusters from biased centroids eventually. In this technique, the researcher does not have many alternatives to deal with this situation: either identifies and excludes outliers (Fischetti, 2015), or he or she assumes that the result is not firmly biased and continues with the analysis.

Fortunately, the development of new clustering techniques in recent years has created consistent options to analyze the effect of outliers. One of these is to use the PAM algorithm, described below.

Technique 2: partitioning around medoids (PAM)

The popularity of *k-means* has stimulated mathematicians and statisticians to develop more appropriate algorithms to deal with the effect of outliers (Pandey & Singh, 2016). One of them is partitioning around medoids (PAM) (Kaufman & Rousseeuw, 1990).

We explained earlier that *k-means* rests on the concept of the centroid, which is a mean. The PAM algorithm starts from a different point, where the medoids are the parameters to form the groups. The medoid is an object allocated to a specific cluster and has the lowest average dissimilarity between it and the others in that cluster, being its most central point (Bhat, 2014; Velmurugan & Santhanam, 2010). Thus, *k-medoids* clusterization is less sensitive to the presence of outliers, as it does not directly use means to define groups (Kassambara, 2017).

The PAM algorithm operates this feature by identifying a *k-number* of representative centroids among the dataset observations. After that, PAM forms clusters by assigning each observation to the nearest medoid. Then this algorithm makes successive exchanges between the medoids and their complementary objects, calculating various objective functions, and comparing their values. When a given exchange corresponds to the smallest possible value of its respective objective function, the algorithm interrupts processing and displays the final result (Kaufman & Rousseeuw, 1990).

We computed the PAM by estimating the optimal number of clusters firstly. Again, we used the elbow and average silhouette methods, operated by a code almost identical to the previous one: the difference is the inclusion of the PAM parameter instead of *k-means*.

```
R> fviz_nbclust(ConsumerScaleSample, pam, method =
"wss") + geom_vline(xintercept = 2,
linetype = 2) + labs(subtitle = "Elbow method")

R> fviz_nbclust(ConsumerScaleSample, pam, method =
"silhouette")
 + labs(subtitle = "Average silhouette method")
```

Figure 4 displays the output of the elbow method.

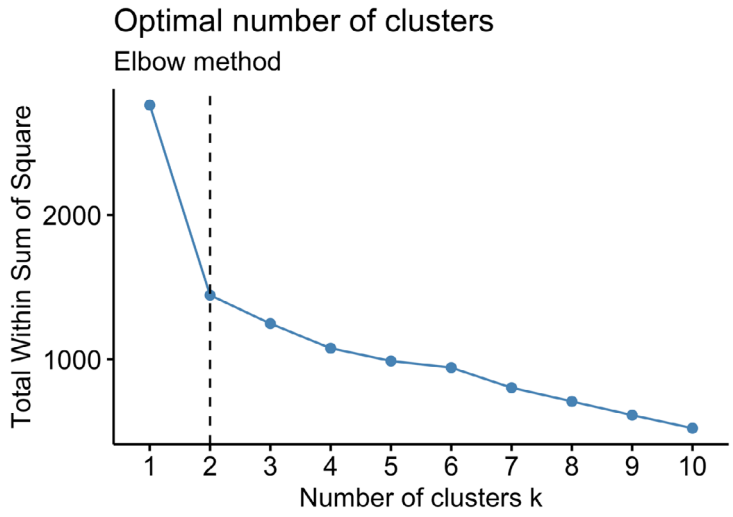

Figure 4. The optimal number of clusters: *k-medoids* clustering with PAM algorithm and elbow method.
Figure 4 has the same interpretation of Figure 1. Hence, the dotted line indicates the optimal number of clusters. That line stands in the point from which the changes in the number of clusters (x-axis) correspond to an incremental variation on the total within sum of square (y-axis). Source: research data.

Figure 4 showed that the use of medoids has not altered the optimal number of clusters that we had estimated in the *k-means* technique. The silhouette method reinforced this conclusion, as shown in Figure 5.

In view of this, we computed the PAM with $k = 2$ and plotted the output in Figure 6.

```
R> pam.Consumer <- pam(ConsumerScaleSample, 2)

R> fviz_cluster(pam.Consumer, data = ConsumerScaleSample,
palette = "jco",
ellipse = TRUE, star.plot = TRUE, repel = TRUE, ggtheme
= theme_minimal())
```

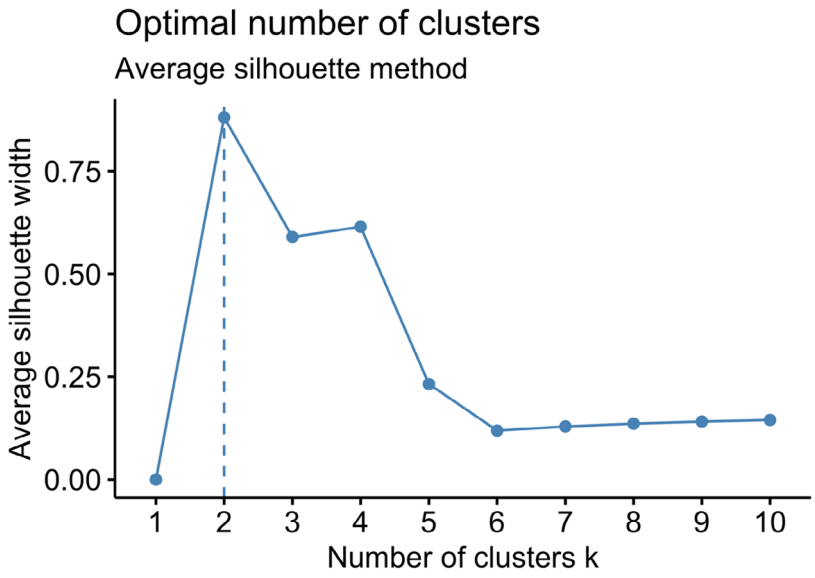

Figure 5. The optimal number of clusters: *k-medoids* clustering with PAM algorithm and average silhouette method.

Figure 5 is interpreted in the same way as in Figure 2. Then, the dotted line indicates the optimal number of clusters according to the average silhouette method. That line stands in the point where there is the number of clusters (x-axis) with the maximum average silhouette width (y-axis). The higher the average silhouette, the better the clustering solution. Source: research data.

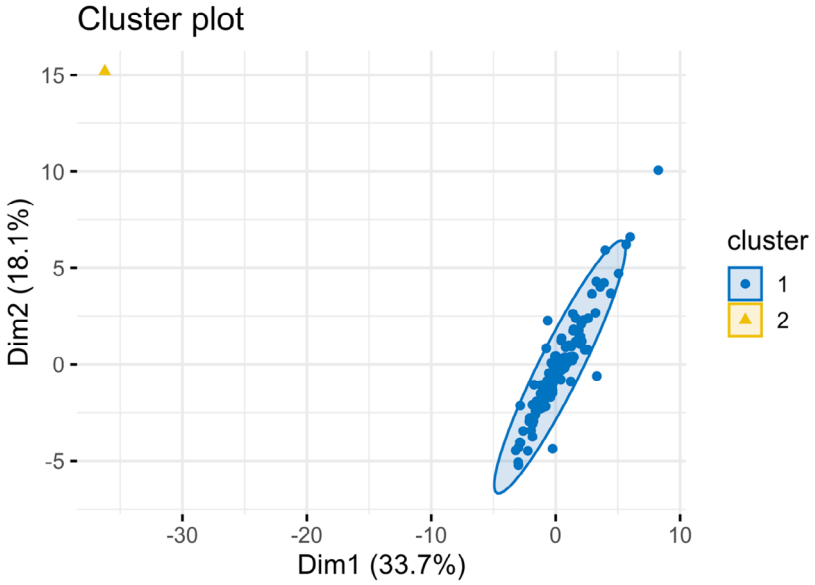

Figure 6. *K-medoids* clustering with PAM algorithm and $k = 2$.

Figure 6 indicates that the PAM algorithm did not change the previous output substantially. Hence, it shows that the consumers have a significant homogeneous behavior considering the variables in the dataset. Because of that, they lied in Cluster 1, except for the one single consumer on the high left corner who was assigned to Cluster 2. Source: research data.

The PAM did not substantially change the *k-means* output. Again, we had Cluster 1 as a convex grouping that included virtually all consumers, except for the one on line 3 of our dataset. That reinforces the hypothesis that outliers are not substantially affecting data behavior. After all, if this hypothesis were the most likely, some evident change in cluster composition should have occurred.

That shows the relevance of the researcher investigating the clustering results further. In our working example, if we had limited ourselves to reporting the *k-means* findings, we might have been tempted to say that the outliers had a significant effect on the data behavior. With PAM, we could practically rule that out.

However, at this point, another question emerged: if outliers are not affecting the allocation of objects in clusters, then would it be better to exclude them from our analysis or treat them as sources of information about data behavior? This question could not be adequately answered by PAM, as it extrapolated its purpose in the same way that assessing the effect of outliers went beyond what *k-means* had to offer. For this reason, we continued our analysis using the third technique: DBSCAN.

Technique 3: density-based spatial clustering and application with noise (DBSCAN)

So far, we have presented techniques that group objects by their similarity in the same cluster, while the clusters differ substantially from each other. For the results to be satisfactory, they need to be spherical-shaped or convex.

However, the researcher can often have objects allocated in clusters of multiple formats. In such cases, none of the techniques discussed so far are suitable. Ester, Kriegel, Sander and Xu (1996) have developed an alternative technique that changes the logic we have adopted so far: the density-based spatial clustering and application with noise (DBSCAN). It classifies objects using density as a parameter, which allows identifying clusters with several geometric shapes.

Density treats clusters as dense and contiguous regions in data space, separated by low-density areas (Sander, 2010). In these areas, unallocated points eventually reside in a cluster: the outliers. That represents a meaningful conceptual change because these points are no longer just anomalous values in a dataset, but become information that can effectively help the researcher understand data behavior. After all, within DBSCAN logic, an outlier is in an area where there is little data concentration, and discovering the cause of this can be a challenge for the researcher.

The first step in using DBSCAN properly is to understand six fundamental definitions (Hahsler, Piekenbrock, & Doran, 2019). The first is the ϵ-neighborhood, given by:

$$N_\epsilon(p) = \{q \in D \mid d(p,q) < \epsilon\}$$

This relationship means $N_\epsilon(p)$ of a data point is the set of points within a specified radius ϵ around p and d is some distance measure. The second definition is point classes. According to it, a point $p \in D$ may assume one of three ways: (a) a core point if $N_\epsilon(p)$ has high density. Therefore, $|N_\epsilon(p)| \geq$ MinPts, which is a user-specified density threshold; (b) a border point if p is not a core point, even though it is in the neighborhood of a core point; (c) a noise point, otherwise.

The third definition states that a point $q \in D$ is directly density-reachable from a point $p \in D$ if two conditions are satisfied: (a) $|N_\epsilon(p)| \geq$ MinPts; (b) $q \in N_\epsilon(p)$.

The fourth definition affirms that a point p is density-reachable from q if there is an ordered sequence of points leading from q to p. The fifth definition states that two points p and q are density-connected if they are density-reachable from a core point $o \in D$. Finally, the sixth definition states that a density-based cluster C is a non-empty subset of D that satisfies two conditions: (a) maximality: $q \in C$, if $p \in C$ and q is density-reachable from p; (b) connectivity: p is density-connected to $q \; \forall \; p,q \in C$.

DBSCAN forms clusters in a relatively simple way. First, the algorithm identifies a core point and allocates it to a cluster with all its density-connected points. Second, this process continues iteratively until it identifies all remaining core and density-connected points. Third, points that have not been allocated to a cluster are the outliers (Kassambara, 2017).

We started DBSCAN by setting $k = 2$ because this value was recurrent in the two previous techniques, by the elbow and average silhouette criteria. Then we provided the value of the *eps* measurement, i.e., the range of the ϵ-neighborhood. For that, we used the following code, creating Figure 7.

```
R> dbscan::kNNdistplot(ConsumerScaleSample, k = 2)

R> abline(h = 2.87, lty = 2)
```

The ideal value of *eps* is at ordinate and is indicated by the dotted line passing through the point of inflection of the curve. However, Figure 7 shows there are two of those points. In that case, it is possible to choose one of them arbitrarily (Starczewski, Goetzen, & Joo Er, 2020). We chose the first inflection point in Figure 7, which corresponds to an *eps* = 2.87 for $k = 2$. We point out that the R algorithm

does not give the *eps* value automatically: it is necessary to run the code several times, bringing the dotted line closer and closer to the first inflection point.

After defining the *eps*, we set MinPts = 3, which is the minimum recommended (Hahsler, Piekenbrock, Arya, & Mount, 2019). Thus, we used the following code, plotting the result in Figure 8:

```
R> db.Consumer <- fpc::dbscan(ConsumerScaleSample, eps
= 2.87, MinPts = 3)

R> fviz_cluster(db.Consumer, data = ConsumerScaleSample,
stand = FALSE, ellipse = TRUE,
show.clust.cent = FALSE, geom = "point", palette =
"jco", ggtheme = theme_classic())
```

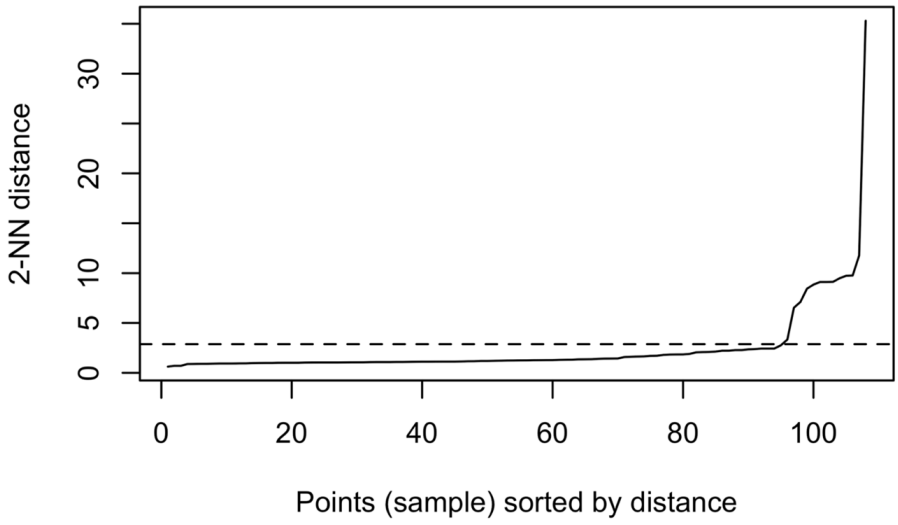

Figure 7. Estimation of optimal eps value in DBSCAN.
The dotted line in Figure 7 lies in the first inflection point of the curve. That point identifies the estimated eps value for the DBSCAN. Source: research data.

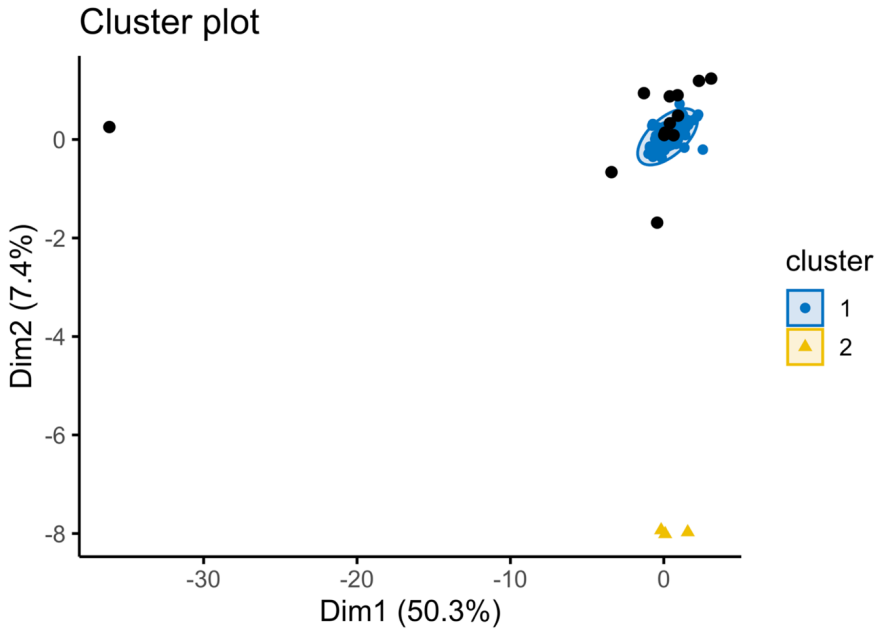

Figure 8. Output of DBCAN clustering with k = 2 and *eps* = 2.87.
Figure 8 indicates that the consumers lied mostly in Cluster 1. Three consumers were assigned to Cluster 2, and this allows us to affirm that they have a distinct behavior from the subjects from Cluster 1. The black dots are the outliers that represent consumers that do not fit in the previous clusters. Source: research data.

Figure 8 showed a different clustering arrangement: instead of a unit cluster, we now had two clusters and outliers indicated by the black dots. They represented consumers with very distinct behaviors from the majority, allocated in Clusters 1 and 2. To identify these consumers, we used the following code:

```
R> db.Consumer$cluster
```

We checked that the outliers were the consumers who were on lines 10, 15, 22, 26, 28, 32, 35, 42, 46, 54, 70, 99, and 103. The rest has formed two groups, i.e., consumers who behaved similarly concerning aspects that affected their overall satisfaction with the firm's products. In this case, researchers could treat the outliers as if they formed a separate cluster and whose components would have relevant information. Consequently, they would not be seen only as a 'problem that needs to be solved,' a common approach in multivariate analysis textbooks (Hair et al., 2018; Malhotra, 2018).

DBSCAN has changed not only the method of clustering but also the very concept and treatment of outliers. As we show in this paper, this technique can broaden the researcher's understanding of data behavior, adding information that would previously be eliminated from the analysis. However, one aspect caught our attention in Figure 8: Cluster 1 comprised two outliers. Although they reside in a low-density area within this cluster, it would be essential to ask ourselves if this allocation would not indicate that we have overlapping clusters. DBSCAN is not appropriate to obtain this answer. The most recommendable in this case is to use a technique specifically developed to deal with this question: fuzzy *c-means* clustering.

Technique 4: fuzzy *c-means* clustering (FCM)

The previous techniques allocate each object to a single cluster. Hence, their algorithms differentiated the objects, so they unequivocally belong to a specific cluster. That is sound logic for the researcher when he works with the assumption that his data will form delimited clusters. However, there may be a second hypothesis: the objects share so many characteristics that these clusters will hardly be utterly separated from each other. In this case, there will be areas of intersection that will change the way for understanding data behavior.

When this second hypothesis is more plausible, the researcher should analyze his data with another clustering technique. After all, *k-means*, PAM, and DBSCAN do not deal with overlapping clusters. A more recommended alternative is to use fuzzy *c-means* clustering (FCM), created by Dunn (1973) and improved by Bezdek (1981). This technique classifies objects by the degree to which they belong to one cluster or another. This degree is measured on a quantitative scale ranging from 0 (low) to 1 (high). In this way, the same object can be allocated to more than one cluster simultaneously, forming an area of intersection.

This aspect is crucial to use the FCM or the other techniques we present in this paper: the researcher must have adequate theoretical support for the research. After all, the mathematical-computational aspect of clustering algorithms exists for the researcher to test this theory. Otherwise, it would be a waste of time to discuss the techniques of quantitative data analysis.

The researcher needs to be careful before choosing FCM. First, he or she needs to rely on a theory that supports the hypothesis that there is such a significant similarity between objects that clusters can overlap. Otherwise, he or she better choose one of the three techniques that we present in this paper. Second, this researcher must understand that FCM does not identify outliers as clearly as DBSCAN, nor does it allow for a specific evaluation of their effect on data, as would occur when comparing *k-means* and PAM results. Therefore, the researcher must keep in mind that outliers may have influenced the formation of clusters by FCM.

We decided to keep $k = 2$ for the same reasons explained above. FCM is part of the *cluster* package that we had installed and loaded previously. Subsequently, we were able to compute it through the code:

```
R> fuz.Consumer <- fanny(ConsumerScaleSample, 2)

R> print(fuz.Consumer)
```

The output demanded 11 iterations, which was within expectations. Moreover, in FCM, it is crucial to analyze the type of cluster formed through Dunn's partition coefficient (*Fk*). It is an indicator that varies from $1/k$ to 1, where values closer to $1/k$ indicate the formation of very fuzzy clusters and values around 1 represent near-crisp clusters. The code displayed both *Fk* and normalized *Fk*, allowing evaluating the output on a scale between 0 and 1.

We had $Fk = 0.5$ and normalized $Fk < 0{,}000$; meaning that there were very fuzzy clusters in the solution; i.e., they overlap significantly. To visualize this, we created Figure 9 with the code below:

```
R> fviz_cluster(fuz.Consumer, ellipse = TRUE,
repel = TRUE, palette = "jco",
ggtheme = theme_minimal(), legend = "right")
```

The spatial distribution of the objects was very similar to those of the previous outputs. However, Cluster 2 included not only the elements previously allocated in Cluster 1 but also those classified as outliers. Moreover, FCM identified an expressive area of intersection between the groups, which was consistent with the value of *Fk* and normalized *Fk*.

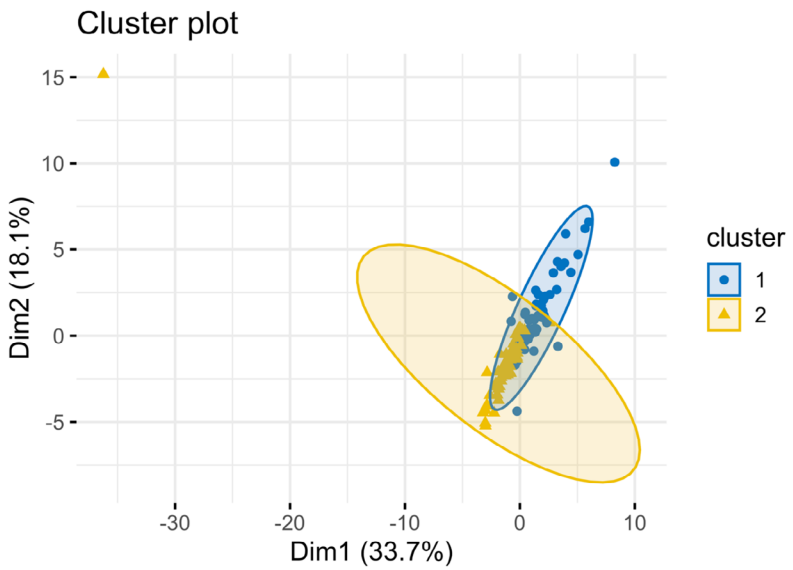

Figure 9. Fuzzy clustering with k = 2. *Fk* = 0.5. Standardized *Fk* < 0.000.

Figure 9 shows that the algorithm identified two clusters that overlap significantly. That output is consistent with the small values of Fk and normalized Fk. The ideal solution would imply higher values for both measures because that would lead to a solution with two or more distinct clusters. Source: research data.

In managerial terms, this would mean that a firm could develop products that would satisfy consumers in both clusters, in addition to maintaining those specific to the other segments. Another option would be to offer a product with characteristics that contemplate the desires of those consumers that are more similar to each other, and that would form the target market of the company.

DISCUSSION OF RESULTS

The clustering techniques we present in this paper represent an opportunity for the researcher to better understand the behavior of the data. After all, the different logics of cluster formation and treatment of outliers, combined with the ease of computing in software such as R, can stimulate the researcher to cross outputs to obtain information that reinforces or weakens the adopted theory.

In this paper, we have seen that there is no more suitable technique a priori. The *k-means* are known and widely used, managing to form groups through a logic easy to understand even by researchers less used to multivariate analysis. Nevertheless, it is sensitive to the presence of outliers in the database, a matter relatively ordinary in research in the managerial area. At the same time, *k-means* does not offer more complex alternatives for the researcher to deal with these discrepant values, other than their exclusion from the dataset. On the one hand, this can be a simple task, since tools such as boxplots usually identify outliers without much computational effort. On the other hand, this can compromise research that uses small datasets, that is, that have up to a hundred objects (Norusis, 2006b).

PAM overcomes these limitations by using a more robust algorithm for the presence of outliers while maintaining the simplicity of the previous technique. It also allows the researcher to evaluate the effect of outliers on data by opposing outputs from other techniques. Hence, PAM avoids that the only alternative of this researcher is to treat the outliers as a problem that needs to be eliminated before starting the central part of the data analysis.

DBSCAN maintains the qualities of the previous techniques by adding an aspect: it not only identifies outliers but also changes the way of approaching them. Consequently, they are no longer necessarily a problem but a potential source of relevant information for decision making. Finally, FCM provides the opportunity for the researcher to find areas of intersection between groups, which can change his understanding of data behavior.

In this paper, we highlight the extent to which each technique has contributed to improving the understanding of data behavior. The *k-means* was the first to indicate that the most likely solution would have only two clusters, formed asymmetrically. Thus, we had Cluster 1 with 34 elements and Cluster 2 with only one. In managerial terms, this would indicate the absence of market segments. After all, there would be a lively group of consumers with similar standards of satisfaction with a given product, and only one individual

differentiating himself from them. In theoretical terms, this solution could indicate that outliers have a substantial effect on data, leading to biased outputs. That could explain why the *k-means* algorithm identified only one convex cluster containing almost all objects.

However, the ease of using other techniques in R allowed us to check the plausibility of these findings with PAM. It provided us with two new pieces of information relevant to the research: (a) the use of a more robust algorithm confirmed the allocation of *k-means* objects, and (b) for this reason, there was not enough reliable evidence to claim that outliers were substantially affecting this output.

DBSCAN was another technique that broadened our understanding of data behavior. It helped us to identify outliers accurately and using the clustering algorithm itself, i.e., without having to resort to external tools like boxplots. Furthermore, its result indicated that the managers of a company could collect additional data from consumers associated with these outliers, to develop products more suitable to their needs eventually.

Finally, FCM showed a different perspective to analyze the data. Clusters could be seen as clusters in which part of the consumers have similar patterns of behavior concerning product satisfaction. Thus, there is an area of intersection that can allow managers to develop products capable of serving two consumer segments simultaneously.

These statements indicate that the researcher may benefit from the use of more than one clustering technique or the choice of one that is most appropriate for the objectives of his research. Table 1 summarizes the advantages and disadvantages of each of these.

The following section presents our conclusion.

Table 1. Advantages and disadvantages of clustering techniques.

Clustering technique	Advantages	Disadvantages
k-means	Simple to compute. Results easy to interpret.	It is sensitive to the presence of outliers in the dataset.
PAM	It maintains the computational simplicity of *k-means*. The algorithm is robust to the presence of outliers.	Different outputs assess the effect of outliers on data behavior.
DBSCAN	The concept of density makes clustering more intuitive. This same concept changes the way of looking at outliers, treating them as information, and not as a problem. Identify outliers graphically.	It does not bring more specific tools for the researcher to evaluate the effect of outliers on data behavior.
FCM	It is less restrictive in the formation of clusters, as it admits that they may be overlapping. It can help the researcher to get relevant insights from this idea of objects belonging to more than one cluster simultaneously.	It does not bring more specific tools for the researcher to evaluate the effect of outliers on data behavior.

Note. Source: research data.

CONCLUSION

In recent years, multivariate analysis methods have become popular in several areas, including management. Among them, one of the most used is clustering, which allows obtaining consistent results even if researchers have small datasets and data that do not follow required distributions in other methods.

However, this apparent simplicity of cluster analysis often leads researchers to believe that no more stringent care is needed with the data. The literature identifies crucial aspects for adequate clustering, such as the scales of the variables and, especially, the effect of outliers. Currently, there are clustering techniques that treat these outliers in different ways, offering alternatives for the researcher that often go unnoticed.

In this tutorial paper, we show how researchers can benefit from the varied repertoire of techniques made available by cluster analysis. Each one of them follows a logic both for the formation of clusters and for the treatment of outliers. That opens two useful alternatives for the researcher both in theoretical and empirical terms.

The first is to choose a suitable technique to assess the plausibility of the theory that underlies the research. In this case, techniques such as *k-means*, PAM, and DBSCAN may be useful if the researcher works with the hypothesis that the objects are probably distinct enough to form separate clusters. Alternatively, the other way around: they are similar to the point where clusters overlap, which would justify the use of a technique like FCM. The second is to analyze the same dataset with different techniques. In this alternative, the researcher

can collect information that will help him to decide to what extent his theory is plausible.

In any of these alternatives, diversifying the repertoire of clustering techniques also means interpreting the outliers differently. In this respect, what we seek to show in this paper is that these data should not be viewed only as problems for analysis, but as relevant sources of information for the researcher. After all, identifying what made them discrepant may be a challenge that will add much to the understanding of data behavior.

REFERENCES

Acock, A. C. (2014). *A gentle introduction to Stata* (4th ed). College Station: Stata Press.

Adams, J., Hayunga, D., Mansi, S., Reeb, D., & Verardi, V. (2019). Identifying and treating outliers in finance. *Financial Management*, *48*(2), 345–384. https://doi.org/10.1111/fima.12269

Aggarwal, C. (2014). An introduction to cluster analysis. In C. C. Aggarwal, C. K. Reddy (Eds.), *Data clustering: Algorithms and applications* (pp. 1-28). New York: CRC Press.

Besanko, D., Dranove, D., Shanley, M., & Schaefer, S. (2016). *Economics of strategy* (7th ed). Toronto: Wiley.

Beysolow, T. (2017). *Introduction to deep learning using R: A step-by-step guide to learning and implementing deep learning models using R*. New York: Apress.

Bezdek, J. (1981). *Pattern recognition with fuzzy objective function algorithms*. New York: Plenum Press.

Bhat, A. (2014). K-medoids clustering using partitioning aroud medoids for performing face recognition. *International Journal of Soft computing, Mathematics and Control*, *3*(3), 1-12. https://doi.org/10.14810/ijscmc.2014.3301

Boehmke, B., & Greenwell, B. (2019). *K-means Clustering* (p. 399–416). New York: CRC Press. https://doi.org/10.1201/9780367816377-20

Caffo, B. (2016). *Statistical inference for data science*. British Columbia, UK: Leanpub.

Dunn, J. C. (1973). A Fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, *3*(3), 32–57. https://doi.org/10.1080/01969727308546046

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996 August). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the International Conference on Knowledge Discovery and Data Mining, Munchen, Germany, 2. Retrieved from https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf

Everitt, B. S., & Hothorn, T. (2006). Cluster analysis. In B. S. Everitt, T. Hothorn, *A handbook of statistical analyses using R* (pp. 243–258). New York: CRC Press.

Fávero, L. P., & Belfiore, P. (2017). Análise de agrupamentos. In *Manual de análise de dados: Estatística e modelagem multivariada com Excel, SPSS e Stata* (pp. 309–378). São Paulo: GEN.

Fischetti, T. (2015). *Data analysis with R: Load, wrangle, and analyze your data using the world's most powerful statistical programming language*. Birmingham: Packt.

Hahsler, M., Piekenbrock, M., Arya, S., & Mount, D. (2019). *Density-based clustering of applications with noise (DBSCAN) and related algorithms*. CRAN. Retrieved from https://cran.r-project.org/web/packages/dbscan/dbscan.pdf

Hahsler, M., Piekenbrock, M., & Doran, D. (2019). dbscan: Fast density-based clustering with R. *Journal of Statistical Software*, *91*(1), 1–30. https://doi.org/10.18637/jss.v091.i01

Hair, J., Black, W. C., Babin, B. J., & Anderson, R. E. (2018). *Multivariate data analysis* (8th ed). Ireland: Cengage Learning EMEA.

Hartigan, J. A., & Wong, M. A. (1979). A K-means clustering algorithm. *Journal of the Royal Statistical Society*, *28*(1), 100–108. https://doi.org/10.2307/2346830

Husson, F., Lê, S., & Pagès, J. (2017). Clustering. In *Exploratory multivariate analysis by example using R* (pp. 173–208). New York: CRC Press.

Irizarry, R. A., & Love, M. (2015). *Data analysis for the life sciences*. British Columbia, UK: Leanpub.

Janssen, A., & Wan, P. (2020). K-means clustering of extremes. *Electronic Journal of Statistics*, *14*(1), 1211–1233. https://doi.org/10.1214/20-EJS1689

Kassambara, A. (2017). *Practical guide to cluster analysis in R unsupervised machine learning*. London: STHDA.

Kaufman, L., & Rousseeuw, P. (1990). Partitioning around medoids (Program PAM). In *Finding groups in data: An introduction to cluster analysis* (pp. 68–125). New York: Wiley-Interscience.

Ketchen, D. J., & Shook, C. L. (1996). The application of cluster analysis in strategic management research: An analysis and critique. *Strategic Management Journal*, *17*(6), 441–458. https://doi.org/10.1002/(SICI)1097-0266(199606)17:6<441::AID-SMJ819>3.0.CO;2-G

Loperfido, N. (2020). Kurtosis-based projection pursuit for outlier detection in financial time series. *The European Journal of Finance*, *26*(2–3), 142–164. https://doi.org/10.1080/1351847X.2019.1647864

Lopes, H. E. G., Pereira, C., & Vieira, A. F. (2009). Comparação entre os modelos norte-americano (ACSI) e europeu (ECSI) de satisfação: Um estudo no setor de serviços. *RAM, Revista de Administração Mackenzie, 10*(1), 161–187. https://doi.org/10.1590/S1678-69712009000100008

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Berkeley symposium on mathematical statistics and probability*, *1*, 281–297. Retrieved from https://projecteuclid.org/download/pdf_1/euclid.bsmsp/1200512992

Maechler, M. (2019). *Package "cluster"*. CRAN. https://svn.r-project.org/R-packages/trunk/cluster

Malhotra, N. (2018). *Marketing research: An applied orientation* (7th ed). New York: Pearson.

Moustaki, I., Jöreskog, K. G., & Mavridis, D. (2004). Factor models for ordinal variables with covariate effects on the manifest and latent variables: A comparison of LISREL and IRT approaches. *Structural Equation Modeling: A Multidisciplinary Journal*, *11*(4), 487–513. https://doi.org/10.1207/s15328007sem1104_1

Norusis, M. J. (2006a). *Cluster Analysis* (pp. 361–391). Upper Saddle River, NJ: Prentice-Hall.

Norusis, M. J. (2006b). *SPSS 15.0 statistical procedures companion*. Upper Saddle River, NJ: Prentice Hall.

Nunnally, J., & Bernstein, I. (1994). *Psychometric Theory*. New York: McGraw Hill.

Pandey, P., & Singh, I. (2016). Comparision between K-mean clustering and improved K-mean clustering. *International Journal of Computer Applications*, *146*(13), 39–42. http://doi.org/10.5120/IJCA2016910868

Peng, R. (2019). *Report writing for data science in R*. British Columbia, UK: Leanpub.

Raykov, Y., Boukouvalas, A., Baig, F., & Little, M. (2016). What to do when k-means clustering fails: A simple yet principled alternative algorithm. *PLoS ONE*, *11*(9), 1–28. https://doi.org/10.1371/journal. pone.0162259

Sander, J. (2010). Density-based clustering. In *Encyclopedia of Machine Learning* (pp. 270–273). Berlin: Springer-Verlag.

Scoltock, J. (1982). A survey of the literature of cluster analysis. *The Computer Journal*, *25*(1), 130–134. https://doi.org/10.1093/comjnl/25.1.130

Starczewski, A., Goetzen, P., & Joo Er, M. (2020). A new method for automatic determining of the DBSCAN parameters. *Journal of Artificial Intelligence and Soft Computing Research*, *10*(3), 209–211. https://doi.org/10.2478/jaiscr-2020-0014

Sugar, C. A., & James, G. M. (2003). Finding the number of clusters in a dataset. *Journal of the American Statistical Association*, *98*(463), 750–763. https://doi.org/10.1198/016214503000000666

Sun, L., Chen, G., Xiong, H., & Guo, C. (2017). Cluster analysis in data-driven management decisions. *Journal of Management Science and Engineering*, *2*(4), 227–251. https://doi.org/10.3724/SP.J.1383.204011

Thrun, M. (2019). Cluster analysis of per capita gross domestic products. *Entrepreneurial Business and Economics Review*, *7*(1), 217–231. https://doi.org/10.15678/EBER.2019.070113

Velmurugan, T., & Santhanam, T. (2010). Computational complexity between k-means and k-medoids clustering algorithms for normal and uniform distributions of data points. *Journal of Computer Science*, *6*(3), 363–368. Retrieved from http://www.thescipub.com/pdf/10.3844/jcssp.2010.363.368

Yu, H., Wang, X., Wang, G., & Zeng, X. (2020). An active three-way clustering method via low-rank matrices for multi-view data. *Information Sciences*, *507*, 823–839. https://doi.org/10.1016/j.ins.2018.03.009

Authorship

Humberto Elias Garcia Lopes*

Pontifícia Universidade Católica de Minas Gerais, Programa de Pós-Graduação em Administração
Av. Itaú, nº 525, Jardim São José, 30535-012, Belo Horizonte, MG, Brazil.

E-mail address: heglopes@gmail.com

https://orcid.org/0000-0002-6207-2726

Marlusa de Sevilha Gosling

Universidade Federal de Minas Gerais, Faculdade de Ciências Econômicas
Av. Antônio Carlos, nº 6627, Pampulha, 31270-901, Belo Horizonte, MG, Brazil.

E-mail address: mg.ufmg@gmail.com

https://orcid.org/0000-0002-7674-2866

* Corresponding Author

Authors' Contributions

1st author: conceptualization (lead); data curation (lead); formal analysis (lead); funding acquisition (lead); investigation (lead); methodology (lead); project administration (lead); resources (lead); software (lead); supervision (lead); validation (lead); visualization (lead); writing-original draft (lead); writing-review & editing (lead).

2nd author: conceptualization (supporting); data curation (supporting); formal analysis (supporting); funding acquisition (supporting); investigation (supporting); methodology (supporting); project administration (supporting); resources (supporting); software (supporting); supervision (supporting); validation (supporting); visualization (supporting); writing-original draft (supporting); writing-review & editing (supporting).

Funding

The authors would like to thank Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq (process #442900/2014-7) and Fundação de Amparo à Pesquisa do Estado de Minas Gerais (process #APQ-03474-17) for the financial resources.

Conflict of Interests

The authors have stated that there is no conflict of interest.

Peer Review Method

This content was evaluated using the double-blind peer review process. The disclosure of the reviewers' information on the first page is made only after concluding the evaluation process, and with the voluntary consent of the respective reviewers..

Plagiarism Check

The RAC maintains the practice of submitting all documents approved for publication to the plagiarism check, using specific tools, e.g.: iThenticate

Data Availability

All data and materials were made publicly available through the Harvard Dataverse platform and can be accessed at:

Lopes, H. E. G., & Gosling, M. S. (2020). Replication data for: Cluster analysis in practice: Dealing with outliers in managerial research. *Harvard Dataverse*, V1. https://doi.org/10.7910/DVN/CN9BEU

APPENDIX A

The More, the Better? When Increasing the Number of Clusters Might Not Be the Best Path

INTRODUCTION

In our paper, we described how four clustering techniques dealt with outliers. The first two of those techniques – *k-means* and *k-medoids* – counted on specific methods for estimating the optimal number of clusters (*k*): elbow and the average silhouette methods. The third technique was DBSCAN that forms clusters based on the points' concentration in a given dataset. The fourth technique was fuzzy clustering that frequently assigns a point into different clusters.

The techniques we described in our paper are second-generation clustering algorithms. It means they are robust enough to present consistent solutions in the first run. That represents a significant change concerning the first-generation techniques presented by widely used textbooks in the managerial area like Hair, Black, Babin and Anderson (2018) and Malhotra (2018). Those techniques do not usually comprise algorithms capable of estimating the best possible solution, demanding researchers to choose the ideal number of clusters heuristically. Hence, those researchers have to deal with the risk of finding a non-optimal solution.

Fortunately, that risk is significantly low in second-generation techniques. The elbow and average silhouette methods give researchers the ideal number of clusters. The DBSCAN identifies groups from a perspective that dismiss researchers of the task of estimating *k* a priori. The fuzzy clustering algorithm iterates data until finding the best solution. Consequently, in those techniques, researchers could have the same outcome even if they changed *k* arbitrarily.

We illustrate that in this Appendix by re-running our paper's code with an arbitrary *k* = 5. The reader will see that we had some different outputs, but the outcomes remained the same. Hence, in practical terms, changing *k* from 2 to 5 did not affect the solutions we presented in the paper.

THE EFFECT OF USING A HIGHER NUMBER OF CLUSTERS

Firstly, we are going to re-run the *k-means* clustering by changing the *k* previously estimated in our paper. Therefore, instead of keeping the *k* = 2 that came from the elbow and average silhouette methods, we will try an alternative and arbitrary one: *k* = 5. The code was:

```
R> km.ConsumerApp <- kmeans(ConsumerScaleSample, 5,
nstart = 30)

R>fviz_cluster(km.ConsumerApp,data=ConsumerScaleSample,
palette = "jco",
geom = "point", ellipse.type = "norm",
repel = TRUE, ggtheme = theme_minimal())
```

Our k-means solution would be like that.

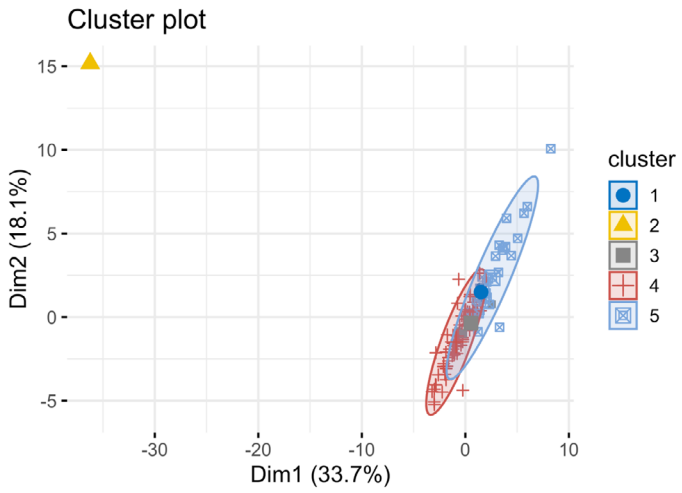

Figure A1. *K-means* clustering with k = 5.

Figure A1 shows that changing *k* from 2 to 5 resulted in four overlapped clusters that only contribute to making the analysis ambiguous and complex. Source: research data.

We can see that *k* = 5 was not useful for improving our previous outcome. Besides, it generated a confusing output where four out of five clusters overlap and, therefore, do not allow us to describe the data behavior precisely.

The same situation occurred in the *k-medoids*. We used the following code to display the solution in Figure A2:

```
R> pam.ConsumerApp <- pam(ConsumerScaleSample, 5)

R>     fviz_cluster(pam.ConsumerApp,     data     =
ConsumerScaleSample, palette = "jco",
geom = "point", ellipse.type = "norm", repel = TRUE,
ggtheme = theme_minimal())
```

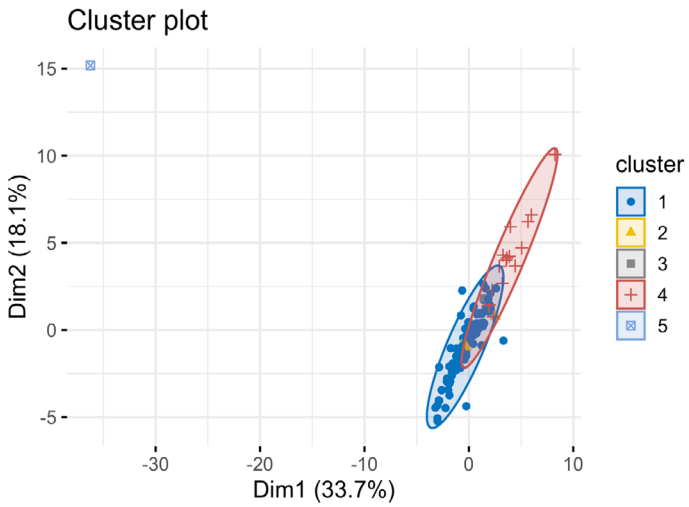

Figure A2. *K-medoids* clustering with PAM algorithm and $k = 5$.

Figure A2 shows that changing k from 2 to 5 resulted in another confusing solution. Once more, we have four overlapped clusters that do not ease the analysis for the researcher. Source: research data.

The output displayed by Figure A2 is practically identical to the previous one. That means we once again generated a non-optimal solution where researchers have to work hard to make any relevant conclusion upon the data behavior.

Those examples indicate that researchers might face substantial difficulties by changing the optimal value of *k* estimated through the elbow and the average silhouette methods. But what if they choose clustering techniques with no support of a specific algorithm to estimate the optimal *k*? We can start answering that question with the third clustering technique in our paper: DBSCAN. As we discussed in our paper, that technique identifies clusters differently from *k-means* and *k-medoids* where the critical parameters are the MinPts and ϵ , not *k*.

Once again, we let $k = 5$, but that value did not imply changes in the MinPts and ϵ. We used the following code to display the output in Figure A3:

```
R> dbscan::kNNdistplot(ConsumerScaleSample, k = 5)

R> abline(h = 2.87, lty = 2)
```

The values of MinPts and ϵ generated the same output we had previously. We used this code for creating Figure A4 and displaying that output:

```
R> fviz_cluster(db.ConsumerApp, data = ConsumerScaleSample,
stand = FALSE, ellipse.type = "norm",
show.clust.cent = FALSE, geom = "point", palette = "jco",
ggtheme = theme_classic())
```

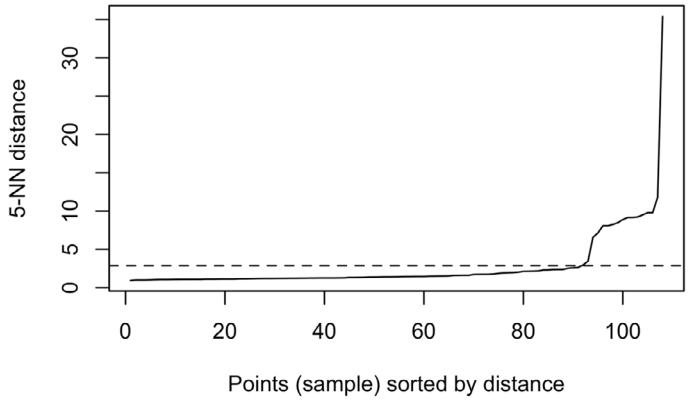

Figure A3. Estimation of optimal *eps* value in DBSCAN.

The dotted line in Figure A3 lies in the first inflection point of the curve. That point identifies the estimated *eps* value for the DBSCAN. That output with $k = 5$ is identical to the one we obtained with $k = 2$. Source: research data.

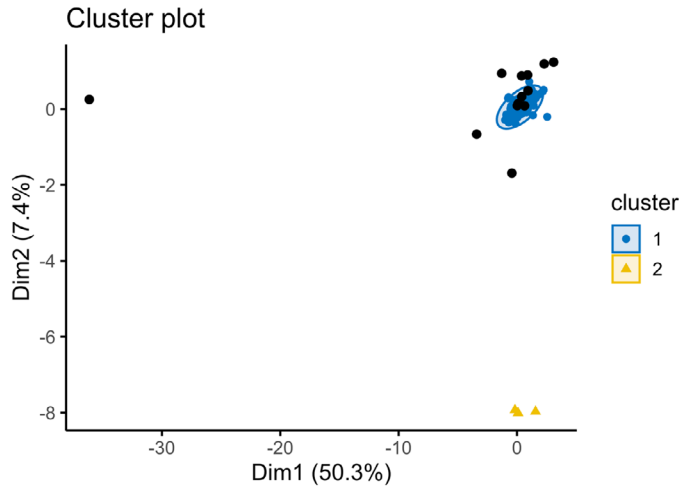

Figure A4. Output of DBCAN clustering with $k = 5$ and $eps = 2.87$.

Figure A4 indicates that the consumers lied mostly in Cluster 1. Three consumers were assigned to Cluster 2, and this allows us to affirm that they have a distinct behavior from the subjects from Cluster 1. The black dots are the outliers that represent consumers that do not fit in the previous clusters. We had that same output with $k = 2$. Source: research data.

The DBSCAN output indicates that changing the number of clusters does not always imply a better description of data behavior. As we affirmed in our paper, DBSCAN uses a different logic for clustering, forming groups based on the data points' concentration. Although we used $k = 5$ that concentration remained the same, as well as the values of MinPts and ϵ.

Lastly, we checked the effect of a higher k on the fuzzy clustering. We used the following code to display the output in Figure A5:

```
R> fuz.ConsumerApp <- fanny(ConsumerScaleSample, 5)

R> fviz_cluster(fuz.ConsumerApp, ellipse.type = "norm",
repel = TRUE, palette = "jco",
geom = "point", ggtheme = theme_minimal(), legend = "right")
```

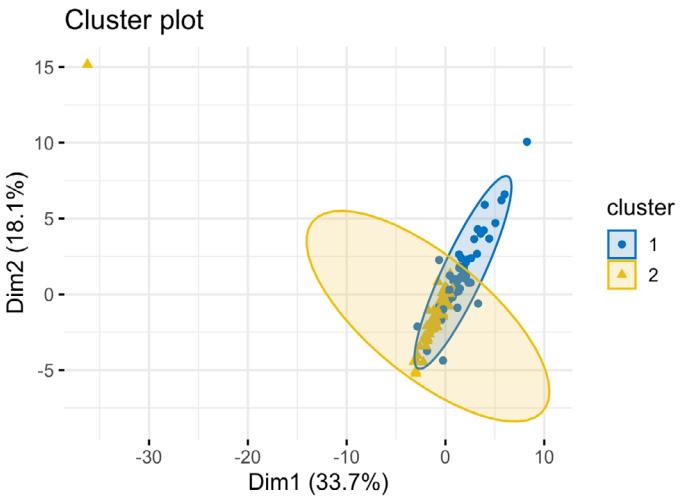

Figure A5. Fuzzy clustering with k = 5. Fk = 0.5. Standardized Fk < 0.000.

Figure 5 shows the algorithm identified two clusters that overlap significantly. That output is consistent with the small values of Fk and normalized Fk.

Once again, k = 5 did not affect the output we had with k = 2. The fuzzy clustering algorithm classified the data points in two clusters only.

CONCLUSION

The outcomes we presented in this Appendix show that the clustering techniques we used in our paper were consistent enough to display an adequate description of the data behavior. Hence, using an arbitrary k did not affect the outcomes. There are two ways of interpreting that statement.

First, it validated, to some extent, the outcomes we obtained in the paper. Since we used the same dataset with n = 108, we did not achieve a better clustering solution by changing k. That was expected for the *k-means* and the *k-medoids* methods since they counted on the elbow and average silhouette algorithms to estimate the optimal k. However, DBSCAN and fuzzy clustering were not based on any pre-defined algorithm for determining k. Despite that, we had the same outcomes in those methods for k = 2 and k = 5.

Second, the outcomes we obtained in this Appendix indicate that the clustering algorithms we used could be very resilient to the researcher's seeking 'adequate' outputs. Unfortunately, some believe that quantitative techniques are useful tools to produce outputs to sustain the assumptions of their research. That misconception is harmful since it spreads the idea that changing parameters at random in a technique is the best way of achieving the desired outcomes.

In this Appendix, we showed that it was no use arbitrarily change k since our paper's outcomes remained the same. That should comfort researchers who are effectively focused on describing the data behavior honestly instead of stalking *ad hoc* evidence to support their assumptions.