Fuzzy C-means: Differences on Clustering Behavior between High Dimensional and Functional Data (Student Abstract)

Carlos Ramos-Carreño
Universidad Autónoma de Madrid
Ciudad Universitaria de Cantoblanco
28049 Madrid Spain
carlos.ramos@uam.es

Abstract

Fuzzy c-means (FCM) is a generalization of the classical k-means clustering algorithm to the case where an observation can belong to several clusters at the same time. The algorithm was previously observed to have initialization problems when the number of desired clusters or the number of dimensions of the data are high. We have tested FCM against clustering problems with functional data, generated from stationary Gaussian processes, and thus in principle infinite-dimensional. We observed that when the data is more functional in nature, which can be obtained by tuning the length-scale parameter of the Gaussian process, the aforementioned problems do not appear. This not only indicates that FCM is suitable as a clustering method for functional data, but also illustrates how functional data differs from traditional multivariate data. In addition this seems to suggest a qualitative way to measure the latent dimensionality of the functional distribution itself.

Introduction

The fuzzy c-means algorithm (FCM) is a clustering method similar to k-means in which each observation, instead of belonging to a particular cluster, has a different degree of membership to each of them. This is a useful property in a number of real-world problems, in which the partition into clusters is not necessary crisp (Nayak, Naik, and Behera 2015).

As in k-means, the algorithm receives as input a dataset $D = \{x_n\}_{n=1}^N$ and the number C of desired clusters. In addition a ω parameter, the fuzzifier, controls the degree of fuzziness of the solution, with its minimum value $\omega = 1$ corresponding to the crisp partition of k-means.

The cluster centers are situated randomly in the initial iteration of the algorithm. The distances d_{ni} between the n-th observation and the i-th cluster center are computed, and the membership matrix U_{ni} between observations and clusters are defined from them as

$$U_{ni} = \left(\frac{d_{ni}^{\frac{2}{1-\omega}}}{\sum_{j=1}^{C} d_{nj}^{\frac{2}{1-\omega}}} \right) \quad n = 1, \ldots, N \quad i = 1, \ldots, C. \tag{1}$$

The cluster centers in the following iterations are computed as $c_i = \frac{\sum_{n=1}^N U_{ni}^\omega x_n}{\sum_{n=1}^N U_{ni}^\omega}$, as the algorithm tries to optimize the inertia function

$$J(D) = \sum_{n=1}^{N} \sum_{i=1}^{C} U_{ni}^\omega d_{ni}^2. \tag{2}$$

In (Winkler, Klawonn, and Kruse 2010), this objective function is expressed as $J(D, \alpha)$, where α is a parameter that determines the initial placement of the cluster centers in several synthetic experiments. The value $\alpha = 0$ corresponds to placing all centers in the center of mass of the dataset, while $\alpha = 1$ corresponds to placing them in their true locations. It is shown that increasing the number C of clusters or the number of dimensions of the data causes a local minimum to appear for low values of α, and a global maximum in the middle. Thus, for initializations on the left side of the maximum, the algorithm converges towards the center of mass and the data is not properly clustered.

The Functional Case

In functional data analysis (FDA) a observation x_n is a function or curve, belonging to an infinite-dimensional vector space such as the Hilbert space L^2 (Ramsay and Silverman 2005). Although a functional observation is often represented inside a computer as discretized evaluations in a grid of size M, one typically should not consider it as a vector of M dimensions in the analysis. This is because the functions of interest are continuous and sometimes smooth, so there is a correlation structure between the discretized values that is different from the one present in multivariate data.

Given the special nature of functional data, it was not clear if the local minimum would appear in the objective function in this case. To test this hipothesis, we have tried to adapt the synthetic datasets used in (Winkler, Klawonn, and Kruse 2010) to the functional case. In particular we considered that our cluster centers are instances of a Gaussian process with a stationary covariance function, such as the exponential kernel $K(x, y) = \sigma^2 \exp\left(-\frac{\|x-y\|}{l}\right)$, with $\sigma, l > 0$. In these covariance functions, the length-scale parameter l controls the amount of covariance between nearby points. Thus, a small length-scale would correspond to curves that change quickly, obtaining in the limit towards 0 the multivariate

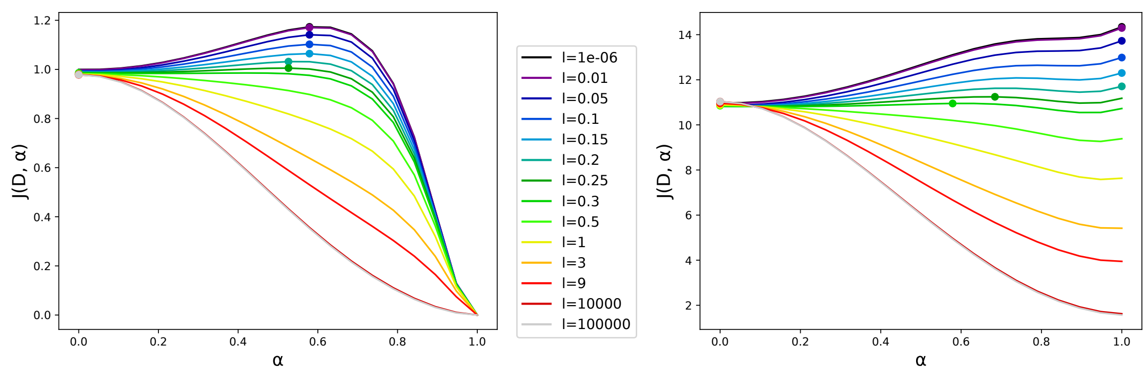

Figure 1: Objective function of FCM as a function of the parameter α, for cluster centers generated as a normalized Gaussian process with a exponential covariance function and different values for the length-scale. In both plots, $C = 50$ and $M = 100$. On the left plot, there is one sample per cluster, while on the right we have 10 samples per cluster, generated from the same Gaussian process centered at the cluster center with a small variance $\sigma = 0.1$. A circle marks the maximum of each curve.

case. A larger length-scale correspond to smoother, more functional curves. Several datasets have been made to consider separately the cases where the cluster centers are normalized (and thus they are placed in a hypersphere, an ideal case for FCM, or they are not (resembling a real dataset). In addition, each of them has two variants, considering the case with only one observation per cluster (an ideal case with no cluster dispersion) and with several samples per cluster, again with a Gaussian distribution.

We have studied the individual influence on the objective function of varying the length-scale parameter l used to generate the clusters, the number C of clusters, the number N of samples (when there is more than one per cluster), the number M of discretization points, and the dispersion when there is more than one sample per cluster. The value of the fuzzifier is fixed at 2 in these experiments.

Results

When we fix the remaining parameters and vary the length-scale l, the multivariate behaviour is replicated for small values of l, as observed in Figure 1. However, when we increase the value of l, the data becomes more functional and the local maximum ends dissapearing. This behaviour has been observed for all synthetic datasets, even for different stationary covariance functions.

For a fixed l the local minimum at the center of mass can appear as the number of clusters increases, and the maximum moves to the right of the plot, making more difficult to obtain a good random initialization. For higher length-scales, however, the number of clusters necessary for the minimum to appear is too high to be a concern in practice. It seems then that FCM works well for functional data, and thus we have included this implementation in our Python software package for FDA, **scikit-fda** (Ramos-Carreño et al. 2022).

Increasing the number of samples for a given cluster has the effect of raising the right minimum, up to a limit that depends on the C, l and the dispersion inside a cluster (σ). For a very high number of clusters and very small length-scale, this can cause the local minimum at the center of mass to be the global minimum, guaranteeing that FCM never obtains a good clustering, as seen in the right plot of Figure 1.

When the number M of discretization points is small, the correlations of the points are small and thus the data behaves like multivariate data of M dimensions. However, as M increases the functional nature of the data becomes apparent, and the objective function essentially stops changing. Thus, the behaviour is no longer the one of a M dimensional vector, and can be considered as if the functional data has some kind of intrinsic dimensionality instead. This idea could be used in the future to obtain a qualitative measure of the dimensionality of the data generated by a functional distribution by comparing it to the multivariate case, which would be useful in the study of functional dimensionality reduction.

Acknowledgments

The author acknowledge financial support from project PID2019-106827GB-I00 / AEI / 10.13039/501100011033 and grant FPU18/00047 from the Spanish Government.

References

Nayak, J.; Naik, B.; and Behera, H. S. 2015. Fuzzy C-Means (FCM) Clustering Algorithm: A Decade Review from 2000 to 2014. In Jain, L. C.; Behera, H. S.; Mandal, J. K.; and Mohapatra, D. P., eds., *Computational Intelligence in Data Mining - Volume 2*, Smart Innovation, Systems and Technologies, 133–149. New Delhi: Springer India. ISBN 978-81-322-2208-8.

Ramos-Carreño, C.; Torrecilla, J. L.; Carbajo-Berrocal, M.; Marcos, P.; and Suárez, A. 2022. Scikit-Fda: A Python Package for Functional Data Analysis. arXiv:2211.02566.

Ramsay, J.; and Silverman, B. W. 2005. *Functional Data Analysis*. Springer Series in Statistics. New York: Springer-Verlag, second edition. ISBN 978-0-387-40080-8.

Winkler, R.; Klawonn, F.; and Kruse, R. 2010. Fuzzy C-Means in High Dimensional Spaces. *International Journal of Fuzzy System Applications*, 11.