You can teach an old dog new tricks: a new algorithm to clustering for big data

by

Cristian Silva

A thesis submitted in conformity with the requirements
for the MSc in Economics, Finance and Computer Science

University of Huelva & International University of Andalusia

Noviembre 2016

You can teach an old dog new tricks: a new algorithm to clustering for big data

Cristian Silva

Máster en Economía, Finanzas y Computación

Manuel Emilio Gegundez Arias
Universidad de Huelva

2016

Abstract

The purpose of this paper is to introduce a new algorithm for numerical classification or clustering. The object of cluster analysis is to classify experimental data in a certain number of sets where the elements of each set should be as similar as possible and dissimilar from those of other sets. Literature provides a wide range of methods for clustering based in a measure of distance or similarity between the elements to be classified, based on the geometry of the full sample. In this paper by contrast, we provide a new algorithm, which offers a second reclassification on the basis of the principal components underlying in each cluster. The work also includes a synthetic example as illustration.

JEL classification: C18; C38; C88.

Key words: summarisation; cluster analysis; classification methods; data mining; principal components.

Resumen

El objetivo de este trabajo es introducir un Nuevo algoritmo para la clasificación numérica o clustering. El objeto del análisis clúster no es otro que el de clasificar datos experimentales en un

cierto número de grupos donde los elementos de cada uno de los grupos deben ser lo más similares posible entre ellos y lo más alejados posible con respecto a los de otros grupos. La literatura proporciona un amplio abanico de métodos para el clustering basados en diferentes métricas de distancia o similitud entre los elementos a clasificar, basados en la geometría de toda la muestra. En este trabajo, por el contrario, proponemos un nuevo algoritmo que ofrece una segunda reclasificación atendiendo a la proximidad a los componentes principales subyacentes al resto de los grupos distintos al de su clasificación inicial. El trabajo se completa con un ejemplo sintético a modo de aplicación.

Tabla de Contenidos

Lista de Figuras

Lista de Apéndices

1 Introducción

Como bien es sabido, el análisis masivo de datos, o la popularizada expresión inglesa de "Big Data" se está convirtiendo en el motor de cambios radicales en diferentes ramas del conocimiento científico, no sólo al permitir explorar y contrastar hipótesis de forma más robusta sino porque su uso está suponiendo cambios metodológicos profundos en un buen número de disciplinas.[1]

Así, y centrándonos en el ámbito de las disciplinas económicas la importancia del procesamiento y análisis de datos a gran escala, está permitiendo no sólo explorar elementos y cuestiones no abordadas hasta ahora, sino cambios más profundos tales como la forma de abordar el estudio de los consumidores, las nuevas posibilidades para la economía computacional y para la predicción financiera o para la toma de decisiones estratégicas. Estos cambios son de tal calado que frente a los tradicionales problemas derivados de la escasa dimensión de las muestras hemos pasado a tener el problema contrario, de forma que el desarrollo de nuevos métodos de sumarización –término con el que se conoce la producción de estadísticos que permiten reducir la dimensión de los datos de cara a su estudio e interpretación– se ha convertido en una de las líneas de trabajo más intensamente exploradas en el ámbito de la minería de datos.

Sin embargo, y no menos importantes están siendo los cambios en las propias técnicas y aproximaciones metodológicas, generando nuevos desarrollos para la manipulación y análisis de datos. Así, la estadística y econometría convencional también necesitan introducir cambios para adaptar sus instrumentos a las nuevas cuestiones que plantea el análisis masivo de datos –nos referimos a la disposición de instrumentos más poderosos para la manipulación de datos, criterios para discriminar entre variables que puedan ser empleadas como predictores alternativos, o la exploración de modelos más complejos (no lineales) gracias a la disposición de

[1] La nueva capacidad de generación a través de diferentes tipos de dispositivos y de almacenamiento de datos es considerada por algunos como una auténtica nueva revolución en el ámbito científico. Baste consultar los trabajos de Marx (2013) o de Varian (2012) para hacerse una idea de la importancia de estos cambios para el conocimiento científico general y para el ámbito económico, respectivamente.

bases de datos mucho más grandes que nos permiten relajar el supuesto de linealidad a favor de estructuras más flexibles (véase, Varian, 2013)–.[2]

Como ejemplo de estos cambios, pensemos por ejemplo en la tradicional forma de abordar los estudios en base a muestreo propias de un buen número de disciplinas en las Ciencias Sociales. La necesidad de arbitrar operaciones estadísticas de elevado coste, con el objetivo de alcanzar una representatividad mínima, y con problemas relacionados con las respuestas estratégicas o subjetivas de los entrevistados, han dado paso a conjuntos de datos basados en conductas observadas y de tal tamaño, que nos permiten el poder seleccionar de manera aleatoria muestras para el análisis a las que podemos realizar pruebas de consistencia y de filtrado y limpiado gracias a un análisis exploratorio previo.

Por todo ello, las nuevas necesidades y características de los datos están provocando que los desarrollos en análisis de datos se tengan que ir adaptando a estas nuevas realidades. En esta evolución la minería de datos, el aprendizaje automático y la econometría están convergiendo para completar la batería de técnicas con las que abordar los diferentes vectores que configuran el análisis de datos[3]. De entre éstos la necesidad de reducir la dimensión de los datos, de proporcionar información resumen –sumarizar–, se hace cada vez más evidente, y de entre estas técnicas, las de clasificación, se tornan esenciales para poder hacer un aprovechamiento óptimo de la información de que se dispone.

En particular, y como ejemplo de su aplicabilidad, la disponibilidad de datos masivos acerca de clientes, asegurados, o empresas, las técnicas de clasificación se convierten en esenciales para al menos clasificar a clientes y empresas según diferentes grados de riesgo y potencial insolvencia, a segmentar a consumidores y clientes para ofrecerles diferentes tipos de productos y diferentes condiciones.

---

[2] En esta línea, hay quien apuesta por una transición desde los tradicionales métodos econométricos basados en las reglas de inferencia clásica hacia técnicas de aprendizaje automático. Nos referimos a técnicas tales como los árboles de decisión, las máquinas de soporte vectorial, las redes neuronales o el aprendizaje profundo, que parecen ser instrumentos más flexibles para modelizar relaciones complejas.

[3] Nos referimos aquí a las técnicas de reducción de datos –sumarización–, la estimación, el contraste de hipótesis y los modelos predictivos.

En este contexto, el reto es aprovechar si la potencialidad de disponer de datos de alta dimensión y frecuencia nos permite diseñar algoritmos alternativos que perfeccionen la clasificación, avanzar hacia la automatización, y en última instancia, que permitan capturar la dinámica de estos *clusters*, no sólo de los individuos transitando entre grupos sino de la propia dinámica de estos grupos.

Este trabajo aborda la primera de estas cuestiones, al proporcionar un nuevo algoritmo de clasificación que no sólo mira en la geometría de la nube total de puntos, sino también en la geometría de un punto con respecto a los componentes principales que definen otros grupos, en un intento de reclasificar puntos que pudieran estar mejor clasificados atendiendo a esos factores determinantes de otros grupos. La intuición que se esconde detrás de esta especie de clasificación en dos etapas puede resumirse con la ayuda del siguiente ejemplo. Supongamos que gracias a las mejoras introducidas en las estadísticas internacionales y gracias a la existencia de nuevas operaciones estadísticas que permiten la comparabilidad, conseguimos disponer de bases de datos de alta dimensión y frecuencia. Supongamos que, como resultado, tenemos un elevado número de indicadores, algunos de ellos muy próximos, que nos llevan a realizar una clasificación basada en muchas variables que sin presentar multicolinealidad se solapan en buena medida en lo que miden. Supongamos que disponemos de un conjunto de macromagnitudes, muchas de ellas alternativas para medir una misma dirección, pero que consideradas en su conjunto pueden generar diferentes tipos de agrupaciones sensibles a la inclusión de todas o algunas de estas variables.

Convendremos que este tipo de situaciones serán altamente probables cuando el Big Data –la alta dimensionalidad y frecuencia– nos ponga en situación de disponer de un elevado número de variables con muy diferente peso en la explicación de la variación total de la base de datos. En estas situaciones, quizás deberíamos chequear si es posible reclasificar algunas observaciones que la clasificación realizada sobre la muestra completa introduce en un grupo aunque probablemente estaría mejor clasificada en otro grupo si en vez de tomar toda la información, tomamos la relevante para cada uno de los grupos, por ejemplo la de los componentes principales de cada uno de los grupos inicialmente extraídos.

Los ejemplos de aplicación del método, se pueden resumir en la provisión de nuevas soluciones al problema de segmentación estática (dinámica) de consumidores, asegurados, grupos de riesgo,

inversores o usuarios de servicios: la automatización y la introducción de cierta dinámica son dos líneas que han de propiciar la emisión de señales automáticas cuando un individuo o un grupo de individuos se encuentra cerca de transitar de un segmento o grupo a otro o cuando este grupo se desplaza porque su centroide también lo hace.

El resto del trabajo se estructura en base a cuatro secciones la primera de las cuales es esta introducción. La segunda sección hace un repaso de la literatura de algoritmos de clusterización, en un intento de contextualizar nuestra propuesta en el estado del arte. La tercera de las secciones se dedica a presentar el método, mientras que el cuarto repasa un ejemplo sintético con el que se pretende ilustrar el uso de nuestro algoritmo. La quinta sección, con la que finaliza este trabajo presenta las conclusiones y algunas posibles líneas de extensión del trabajo.

2 El estado del arte

El clustering puede ser considerado como un proceso de aprendizaje no supervisado, que a diferencia de la clasificación, no impone categorías o predefinidas. Supongamos que disponemos de un conjunto de *n* elementos, que deseamos particionar en *k* subconjuntos, grupos o clústers $C_1,C_2,...,C_k$, de forma que los elementos de cada grupo presenten la mayor similitud posible intragrupo y disimilitud entregrupos. Como bien es sabido disponemos de una amplia gama de de algoritmos para realizar la clusterización, cuya elección depende, en buena medida de los objetivos. En cualquier caso, y atendiendo a la forma en la que generamos los clusters, estos algoritmos pueden ser clasificados al menos en las siguientes categorías: *jerárquicos*, *de particionamiento*, *basados en densidad*, métodos basados en el GRID, métodos basados en la co-ocurrencia de datos categóricos, algoritmos de clustering escalable y algoritmos para datos de alta dimensión.

La siguiente tabla resume de forma no exhaustiva estos algoritmos. Aunque es habitual encontrar diferentes categorizaciones de algoritmos de clustering algorithms estas no son canónicas, ya que muchas de ellas se solapan.

Algoritmos de clusterización
1. Métodos jerárquicos
   1.1. Algoritmos aglomerativos
   1.2. Algoritmos Divisive
2. Métodos de particionamiento
   2.1. Algoritmos de relocalización
   2.2. Clustering Probabilístico

2.3. K-medianas
2.4. K-medias
2.5. Algoritmos basados en densidad
2.5.1. Clustering de conectividad basado en densidad
2.5.2. Clustering basado en funciones de densidad
3. Métodos basados en el GRID
4. Métodos basados en la co-ocurrencia de datos categóricos
4.1. Gradiente Descent and Redes neuronales artificiales
4.2. Métodos evolutivos
5. Algoritmos de clustering escalable
6. Algoritmos para datos de alta dimensión
6.1. Clustering de subespacios
6.2. Técnicas de proyección
6.3. Técnicas de Co-clustering

Una propuesta de clasificación de algoritmos de clusterización

Comúnmente, los algoritmos de clustering jerárquico se suelen clasificar en aglomerativos que parte de generar un número cluster equivalente al número de observaciones de la muestra e ir de manera iterativa uniendo en grupos según la medida de similiaridad que usemos. El proceso se realiza iterativamente con el resto hasta obtener un cluster que engloba a todos. Los divisivos, por el contrario, actúan de forma inversa. Partiendo de un único cluster se van separando en grupos más pequeños. De forma general, los divisivos son más ineficientes desde el punto de vista computacional, excepto cuando se tratan de variables binarias. Cuando la base de datos no es demasiado grande el clustering jerárquico obtiene buenos resultados, pero al depender de los resultados por la sección que se corta podemos encontrarnos con resultados peores que con otros métodos.

Otros algoritmos, denominados de partición rápida, buscan de conseguir una partición inicial satisfactoria. De algún modo, son muy útiles para aplicar a otros métodos como el *K-means* donde en ocasiones no sabemos determinar el número de cluster inicial. Los dos grandes grupos de técnicas de agrupación más intuitivos y más extendidos, son los de agrupamiento ya estén basados en centroides (*k-means*, *k-medianas*) y los basados en la conectividad, los llamados métodos de *clustering jerárquico*. El algoritmo *k-means*, también conocido como algoritmo de Lloyd (1982) asigna un conjunto inicial de *k-centroides*,[4] lo que equivale a marcar a priori el número de particiones al conjunto de datos queremos realizar. La asignación de las observaciones a cada centroide equivale a asignar cada observación al *cluster* más cercano tras lo

[4] La misma lógica aplica en el caso del de k-medianas.

cual se recalculan de nuevo los centroides. Aplicando iterativamente esta secuencia, el proceso termina –converge– cuando las nuevas asignaciones ya no generan modificaciones en la clasificación a un grupo. Frente a este método, Hamerly y Elkan (2002) proponen una alternativa para encontrar mejores *clusters* basados en una partición aleatoria. Éste asigna aleatoriamente un clúster para cada observación y después se procede a su actualización.

Por su parte, también existen modelos mixtos que basan su algoritmo en el uso de algún tipo de distribución de probabilidad (Dempster 1977).

Por otro lado, no han sido escasas las variantes del *k-means* que se han propuesto. Por ejemplo el fuzzy c-means, propuesto por Zadeh (1965) y perfeccionado por Bezdek et al. (1984) observa que cada elemento u observación tiene un grado de pertenencia difuso a los grupos. Para conseguir esta similitud entre un elemento y un grupo se logra con una función de pertenencia que toman valores entre 0 y 1, donde valores cercanos a 1 representan mayor similitud. Este algoritmo surge de la necesidad de resolver un problema de agrupamiento exclusivo, donde se considera que cada elemento se agrupa inequívocamente con los elementos de su cluster y da por hecho que no se asemeja al resto de los elementos.

Existen además otros métodos que ayudan a mejorar la elección de los centroides iniciales como es el k-means++ propuesto por Arthur, D. y Vassilvitskii, S. (2007). Su base consiste en encontrar grupos de puntos tal que se minimice la varianza intra-grupos. Otros presentan una implementación sencilla y eficiente del algoritmo k-means donde utilizan una estructura de datos principal kd-tree como es el propuesto por Kanungo et al (2002). Estos mejoran la eficiencia en cada paso del algoritmo por el cuál ha sido usado en segmentación de imágenes.

El otro gran grupo de técnicas de clustering y más utilizado se basan en la conectividad o agrupamiento jerárquico formando un dendrograma en forma de árbol.

Rokach et al (2005) los separó en dos grupos: Clustering aglomerativo (cada observación es un propio cluster y a él se le van uniendo otros clusters vecinos) y clustering divisivos (todas las observaciones forman parte de un único cluster que se va dividiendo en clusters más pequeños). Computacionalmente el clustering aglomerativo es más sencillo pero puede producir problemas de agrupaciones incorrectas al tener un conjunto de datos grande, en cambio los divisivos son complejos computacionalmente pero muestran mejores resultados con grandes datos.

Para cerrar nuestro análisis, los algoritmos también admiten clasificaciones atendiendo a las diferentes métricas o distancias entre pares de observaciones. Así, y en función de la medida de disimilitud con la que se realiza la división, nos encontramos con SLINK que es la medida que contiene el par de elementos más cercanos (Sibson, 1973). Defays (1977) habló del algoritmo usando la medida intercluster máxima o completa, conocida como CLINK. Por último, UPGMA es un algoritmo atribuido a Sokal y Michener (1958) que construye el dendrograma tomando las distancias medias. Una variación de éste último es usar el centroide del cluster en vez de la media aritmética haciéndose llamar UPGMC.

Otros criterios de enlaces son el criterio propuesto por Ward (1963). El criterio de Ward o método de la mínima varianza elige el par de cluster para unir en función de un valor óptimo de una función objetivo. Ward propuso como función objetivo la suma de los errores al cuadrado. Siguiendo el método de Ward se han implementado variaciones de la misma como es el uso de pesos específicos para los clusters (de Amorim, 2015). Finalmente y para datos de alta dimensión Zhang et al (2012) proponen un método basado en dos conceptos "indegree" y "outdegree" como agregación y desagregación, donde usa como medida de afinidad de cluster el producto de las medias de agregación y desagregación.

En los algoritmos divisivos destaca el propuesto por Kaufmann et al (1990) denominado DIvisive ANAlysis Clustering (DIANA). Aquí, inicialmente todas las observaciones están en el mismo cluster, luego se elige la observación con máxima disimilitud media y posteriormente se mueven todas las observaciones que son más similares a ese nuevo grupo.

Una vez repasada la literatura previa, pasemos a abordar los principales elementos de nuestra propuesta metodológica.

3 Metodología

Como hemos advertido, y para aplicar el método de clustering de *k-medias* debemos especificar, previamente, el número deseados de grupos *k* que deseamos tener como resultado de la clasificación. El método, a continuación asigna aleatoriamente uno de los k grupos a cada una de las observaciones, iterando hasta que las observaciones no cambian al encontrar un mínimo local, asignación que se puede realizar atendiendo a diferentes métricas. La eficiencia y sencillez

de este método cuando fijamos el número de *cluster k* lo han convertido en un método muy utilizado, aunque entre sus desventajas se encuentra su dependencia de esta elección.

El algoritmo que desarrollamos tiene dos etapas, una primera en la que aplicamos el método de *k-medias* para un determinado número de grupos fijados por el usuario, al que permitimos tres opciones en función de los parámetros proporcionados. Así, consideraremos un primer caso en el que las particiones se realizan sin indicarle ningún parámetro –proceder de esta forma equivale a realizar el k-medias con la distancia euclídea. Una segunda opción pasaría por se seleccionar qué tipo de distancia queremos usar y una tercera en la que le indiquemos un valor inicial desde el que partir para elegir el centroide del *cluster*.

La segunda parte del código (algoritmo) se va realizando sobre la base de la primera etapa. En esta segunda etapa se crea una estructura formada por:

- Las coordenadas de los centros de cada *cluster*, dato que es importante como referencia de los puntos y que además permite ver la trayectoria en el tiempo de dichos centros. Esta consideración tiene mucho valor de manera que podremos conocer el comportamiento del clúster, qué dirección va tomando y con qué velocidad se mueve, entre otros aspectos.

- Autovalores y autovectores de cada *cluster*. El cálculo de los autovalores y autovectores sigue la idea de obtener las observaciones en base a su propio clúster y así poder determinar cómo se posicionan las observaciones de forma endógena en el clúster. Una nueva dirección y módulo provoca que se puedan tener las distancias reales de las observaciones dentro de su clúster.

- El número de observaciones de cada *cluster* es un valor que aporta información de magnitud, esto es el conocer cuántos individuos hay en cada clúster.

- Las coordenadas de las observaciones con respecto a sus centros. Este elemento es central, dado que cada *cluster* tendrá una forma determinada y por lo tanto, tener sus observaciones centradas nos permite conocer su posición real dentro del *cluster*. Por otro lado, es un paso necesario para obtener las distancias en función de los autovectores y autovalores propios de cada clúster.

- Las distancias en la base canónica, sin tener en cuenta autovectores y autovalores de los cluster, porque nos interesa ver directamente no sólo su distancia euclídea, sino además la distancia en la nueva base con respecto a su centro, con un autovector y autovalor que lo determina. La ventaja de obtener las distancias en la nueva base supone considerar la distribución de todas las observaciones del *cluster* y de este modo el tránsito de *cluster* de las observaciones está sujeto a estas distancias. Las distancias calculadas en el paso anterior son la base sobre la que se asienta la solución al problema planteado, esto es, conocer que observaciones están más alejadas del centroide. Para estimar estas observaciones alejadas nos ayudamos de la desviación estándar, que nos ha de aportar una información completa sobre cómo varían las observaciones y su grado de dispersión en la población. Indicar en este punto, que dejamos a criterio del usuario el valor de desviación que quiere para su conjunto de datos ±1, ±2 o ±3 desviaciones estándar.

- La compacidad o dispersión es una medida de gran valor muy utilizada en sistemas territoriales. El concepto de dimensionar las densidades de cada *cluster* arroja información sobre la estructura de dichos *cluster* y por consiguiente abre un abanico de posibilidades de gestión atendiendo al grado de compacidad. Matemáticamente se resuelve como el producto de los autovalores calculados para cada *cluster* partido por el número de observaciones.

- Por último, se planteó la posibilidad de detectar las observaciones que están en "peligro" de transitar de un *cluster* a otro mediante las distancias respecto a *clusters vecinos*. La idea se fundamenta en localizar las observaciones donde la distancia a *clusters vecinos* es menor que la distancia hacia el centroide de su propio *cluster* y de este modo, consideramos que las observaciones están en peligro. Esto presenta una ventaja previsora en el sentido de poder tomar una decisión con respecto a estas observaciones en peligro de cambio, si se trata de clientes, empresas o consumidores.

Estas estructuras descritas se pueden generar para cualquier periodo dado y que se realiza de forma automática. Sobre la base de esta estructura se hace uso de toda la información proporcionada para obtener de la segmentación de un conjunto de datos.

4 Muestra

Para presentar una aplicación del algoritmo anterior, hemos procedido a generar una muestra que recoja una serie de características de ingresos y gastos mensuales durante un período dado, en vez de utilizar los microdatos de alguna de las encuestas de presupuestos familiares que podemos recopilar para alguna región o país. Así, hemos procedido a generar una muestra sintética que de algún modo pueda ser reflejo de la realidad, y que nos proporcione una variabilidad y riqueza que nos permita apreciar la potencia del algoritmo planteado.[5]

El principal problema de esta simulación es la de cómo contemplar, de manera realista, todos los aspectos que influyen en la dinámica de ingresos y gastos de una determinada población. Este problema se ha tratado de solventar suponiendo que la dinámica de gastos e ingresos de un determinado individuo o unidad económica de consumo pueden agruparse en tres grandes categorías: i) cambios que afectan a las decisiones económicas de oferta de trabajo y de consumo de los hogares, esto es tanto a sus trayectorias de ingresos como de costes; ii) condiciones individuales que podemos considerar invariantes en el tiempo aunque diferentes entre individuos; y iii) factores de entorno macroeconómico, esto es shocks, que afectan a todos los hogares y que varían en el tiempo.

Estas categorías servirían de base para obtener datos de ingresos y gastos de los n hogares/individuos/consumidores a incorporar en la muestra en cada período t .

Entorno macroeconómico: recogemos bajo este epígrafe un conjunto de factores de entorno que afectan de forma común a todos y cada uno de los agentes incluidos en la muestra. La evolución de las variables reales como el crecimiento y el empleo, o nominales como los precios y los tipos de interés, así como los shocks de oferta y demanda sufridos por la economía y por los agentes que en ella participan, son algunas de las variables que tratan de ser capturadas por esta categoría. La inclusión de una tendencia en la función generatriz es la forma en la que procedemos a incorporar de manera global el efecto de este tipo de factores.

---

[5] El uso de esta muestra generada por simulación nos resuelve una serie de problemas habitualmente presentes en otro tipo de muestras aunque somos conscientes de que la introducción de algunos supuestos simplificadores o incluso de algunas funciones generatrices pueden ser demasiado simples. Sin embargo, creemos que esta generación de la muestra es especialmente útil para los objetivos perseguidos en este trabajo.

Variables con variabilidad individual y temporal: para generar estos flujos de ingresos y gastos, se procede a establecer una escala [-5,5] que representa el grado de rendimiento de cada unidad muestral. Además de proceder a clasificar a los individuos con un valor en esa escala de rendimiento, también establecemos una especie de probabilidad de cambio, esto es, la existencia de un factor que puede provocar que este rendimiento pueda cambiar en el tiempo. Así, y estableciendo como supuesto el que esta probabilidad sea menor que 0.02 –para hacerla meramente ocasional– también establecimos el supuesto de limitar el tamaño del cambio en un rango [-2,2] del valor sin llegar a poder ser mayor o menor de la escala [-5,5].

Variables con variabilidad individual pero no temporal: consideramos en esta categoría un conjunto de factores que son diferentes entre individuos pero que son invariantes o relativamente invariables, al menos en el corto plazo. Características individuales tales como el nivel educativo, el género, el intervalo de edad, lugar de residencia, dependientes a su cargo u otras características que determinan la pertenencia a un determinado nivel socio-económico y por tanto una cierta trayectoria en sus flujos de ingresos y gastos. Para capturar esta categoría, se ha procedido a otorgarles un comportamiento aleatorio tanto para un mismo individuo como entre individuos de diferente estatus. Siguiendo con lo comentado, se diseñó computacionalmente este aspecto aplicando una escala de valor entre [-5 y 5] aleatorio para cada individuo y en cada periodo a modo de porcentaje. Al ser una escala negativa y positiva se buscaba que este factor pudiese afectar de una forma positiva y negativa a los ingresos y gastos de las n unidades muestrales consideradas.

Siguiendo estos criterios, procedemos a realizar la simulación de nuestra muestra de hogares y trayectorias de ingresos y gastos. Para el diseño de los ingresos se procede a crear, previamente y de forma aleatoria, dos subgrupos: el de los hogares de ingresos medios y el de los hogares de ingresos altos.

La función generadora de los ingresos para los hogares de la muestra es:

$$ING_{(1:n,i)} = ING_{(1:n,i-1)} + (Tend_{(i)} - Tend_{(i-1)})/(\text{Tend}_{(i-1)}) \cdot ING_{(1:N,i-1)} + ING_{(1:n,i-1)} \cdot F\_lp_{(1:n,i-1)} \cdot \boldsymbol{\alpha} + ING_{(1:n,i-1)} \cdot F\_cp_{(1:n,i-1)} \cdot \boldsymbol{\beta}$$

Donde *ING*, representa la matriz de ingresos, *Tend,* es la tendencia de la serie, F_lp, son las variables invariantes en el tiempo pero variantes entre individuos, F_cp: son los factores que varían entre individuos y en el tiempo, Factores a corto plazo, mientras que α y β son los parámetros de ajuste, siendo *n* el número de hogares e, *i* el período temporal.

Por su parte, y para el diseño de la variable gastos procedemos a obtenerla, inicialmente, en función de los ingresos presuponiendo que éstos han de situarse en un rango aleatorio que suponga entre el 40 y el 80 por ciento de los ingresos de cada período. Procediendo de este modo obtendremos la matriz de gastos para el momento inicial (i-1).

Por analogía, la función generadora de gastos viene dada por:

$$GAST_{(1:N,i)} = GAST_{(1:N,i-1)} + (Tend_{(i)} - Tend_{(i-1)})/(Tend_{(i-1)}) \cdot GAST_{(1:N,i-1)} - GAST_{(1:N,i-1)} \cdot F\_lp_{(1:N,i-1)} \cdot \alpha - GAST_{(1:N,i-1)} \cdot F\_cp_{(1:N,i-1)} \cdot \beta$$

Donde *GAST*, denota ahora a la variable de gasto, y en la que los factores inciden ahora de forma inversa a como lo hacían en la ecuación de ingresos.

Una vez generada la muestra, cuyo código se reproduce en el apéndice A, esta es utilizada para generar una segmentación haciendo uso del método de *K-means,* aunque el código podría adaptarse a otro algoritmo diferente.

5 Experimentación y resultados

En el apéndice se muestran junto al código de generación de la muestra, el código del algoritmo de clasificación escrito en *matlab* que constituye el núcleo de este trabajo, y en el que no sólo nos fijamos en la geometría de la muestra completa, sino también en la geometría de cada grupo o *cluster* para proceder a realizar reclasificaciones sobre la base de esta última.

Llegados a este punto procedamos a presentar la aplicación, el ejemplo sintético, que nos ha de servir para ilustrar cómo funciona el algoritmo en esta especie de clasificación en dos etapas. Para ello, comencemos por analizar la muestra generada tal y cómo se ha descrito en la sección anterior. En la primera de las figuras (figura 1), se representa la función generada para representar las variables de entorno macroeconómico que sirven para generar la muestra de ingresos y gastos de los hogares. Esta función muestra una determinada evolución del ciclo

económico caracterizada por una tendencia y una serie de fluctuaciones periódicas en torno a la misma

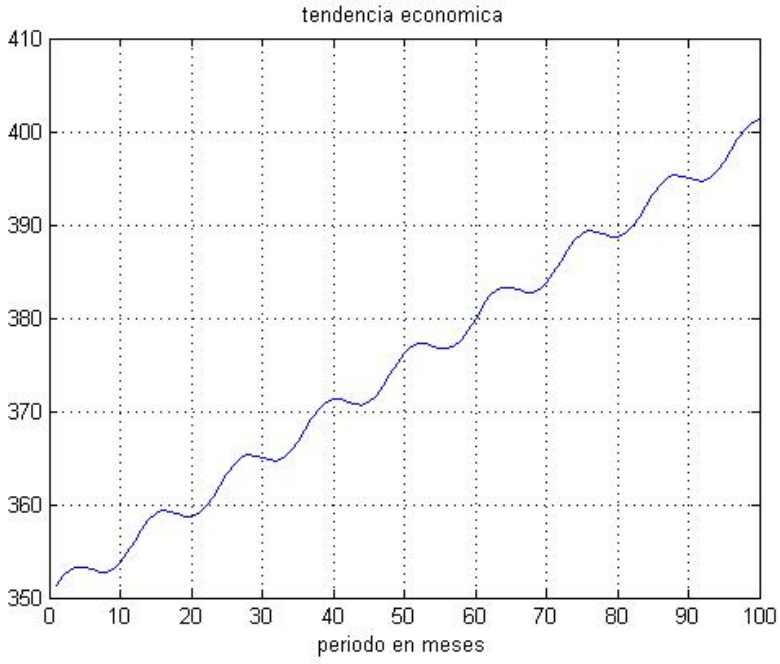

Figura 1. Simulación de la evolución macroeconómica de la economía considerada.

Por su parte, las variables con variabilidad individual pero no temporal –los llamados factores a corto plazo, se han generado a partir de unos datos de entrada –el número de hogares/indviduos y el espacio temporal–. Sobre esta base, hemos generado, de forma automática una matriz donde se generan los valores aleatorios en un intervalo de ±5%.[6] La figura 2 nos muestra la variación de esta variable generada dentro del intervalo mencionado para un periodo de 100 meses.

[6] La elección de este intervalo ±5% no pasa de ser un rango, creemos que aceptable como supuesto general sobre la variabilidad normal en ingresos y gastos de un individuo.

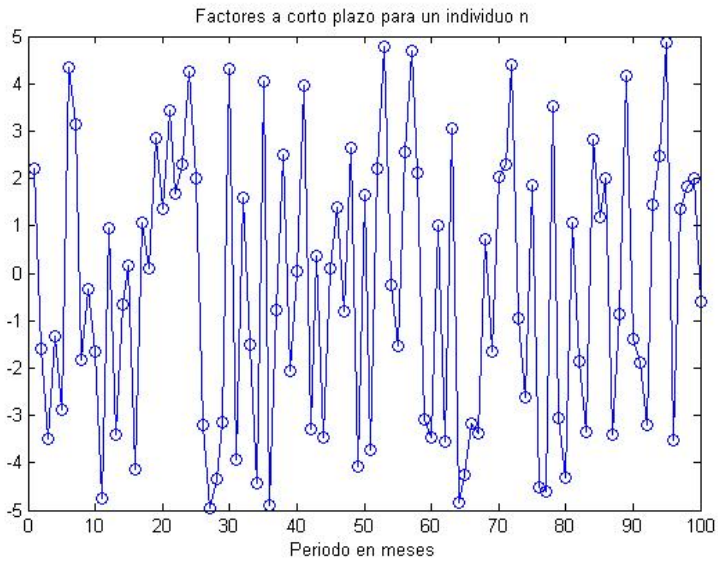

Figura 2. Representación en el tiempo de los factores a corto plazo para un individuo n.

En la figura 2 podemos observar los valores aleatorios que pueden tomar estas variables para un individuo a lo largo de un periodo. Estos valores que se han considerado aleatorios, corresponden a decisiones que pueden ser diarias y éstas tienen una influencia débil. Realizando una división en la interpretación de cómo afecta este factor a los ingresos y gastos concluimos que los factores afectan inversamente. Esto se entiende que cuando tomamos una decisión que nos afecta de forma positiva se ve reflejado en un incremento de los ingresos y en un ahorro en los gastos.

Además de estos factores, recordemos que en el código de generación de la muestra también íbamos a considerar variables que cambiaban entre individuos pero con mucha estabilidad temporal –cambios de empleo, residencia, aumento de familiares …– Se trata pues de cambios que se producen con muy poca frecuencia y a los que le asignaremos una baja probabilidad de ocurrencia. Este parámetro –umbral de cambio–es un parámetro variable y que se ha considerado de 0.02 (2%).

El resultado que se obtiene es una matriz con valores en un intervalo [-5,5] a lo largo de un periodo para todos los individuos. Al igual que ocurre con los factores a corto plazo, si este valor es positivo afecta en un aumento leve en los ingresos y en una reducción en los gastos. Teniendo en cuenta estos tres tipos de variables determinantes del flujo de ingresos y gastos de los hogares, se procede a generar una muestra de hogares tipo.

A modo de ejemplo, y para visualizar el aspecto que tendría una muestra tipo generada sobre la base de las consideraciones anteriores, la figura 3 representa la dinámica de ingresos y gastos para cuatro individuos escogidos de forma aleatoria y para 84 períodos.

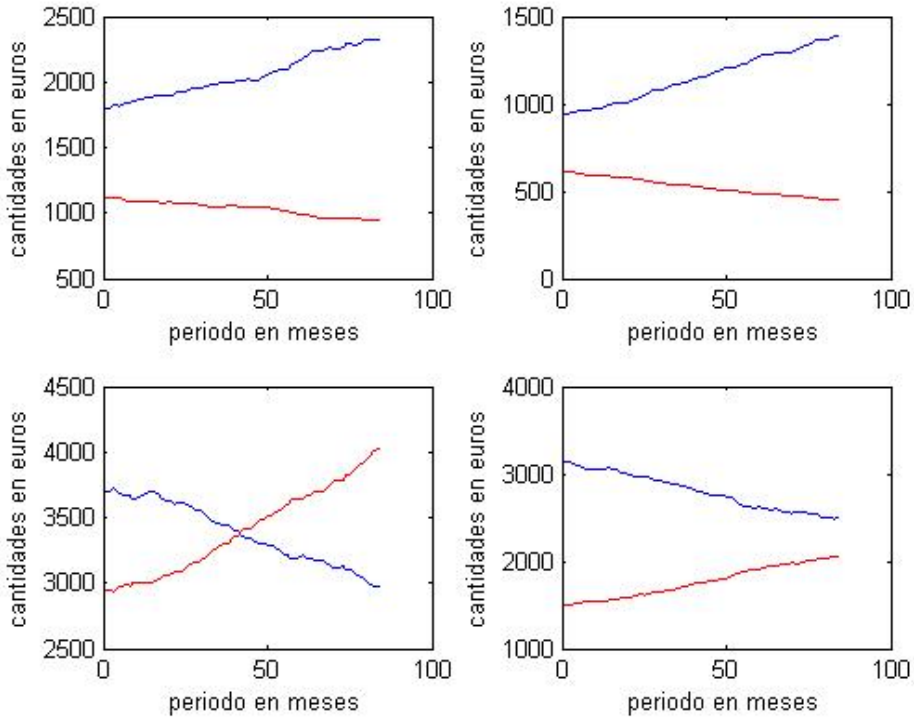

Figura 3. Ingresos y gastos para cuatro individuos diferentes en 84 períodos.

Los resultados que se pueden ver en la Gráfica 3 muestran la evolución de los ingresos (azul) y gastos (rojo) en un período de 84 meses. Cabe destacar las diferentes evoluciones que pueden presentar los individuos a lo largo del tiempo dado que están sujetas a los factores ya comentados.

Llegados a este punto, estamos en disposición de, haciendo uso de esta muestra, proceder a presentar los resultados del mecanismo de clusterización propuesto. Para ello comenzaremos por la clasificación de la muestra inicial de hogares en N-clusters mediante el algoritmo de k-medias, en un período inicial. Dado que la idea es la de generar una herramienta de clasificación dinámica, hemos de hacer una clasificación para cada uno de los períodos. Así pues, el número de grupos, el período y el tipo de distancia a emplear han de ser los parámetros que hemos de decidir para proceder a la clasificación. Junto a estos, también es posible establecer un centroide

inicial, opción muy interesante desde el punto de vista dinámico porque permite realizar un *cluster* entre dos períodos consecutivos, tomando como centroides los generados en el momento anterior. El alcance de esta opción produce una mejor eficiencia del método *k-means* porque de esta forma la dinámica cobra más sentido y ayudamos a que el algoritmo *k-means* no tenga que recalcular los *cluster* en cada momento.

Los resultados gráficamente mostrarían al conjunto de datos agrupados en los grupos que se hayan considerado para cada momento del periodo. Un ejemplo tipo que hemos usado es tomando una partición en dos grupos y capturar la posición que tienen los individuos para un diferentes períodos, en particular los períodos 1, 15, 30, 45, 60 y 75.

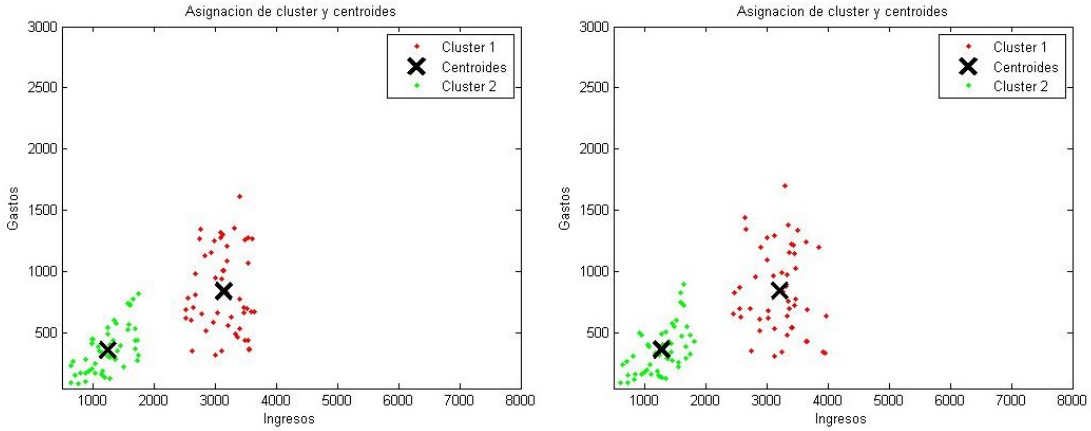

Figura 4. Asignación de cluster y centroides para una población dada para el período 1 (Panel izquierdo) y el 15 (Panel derecho).

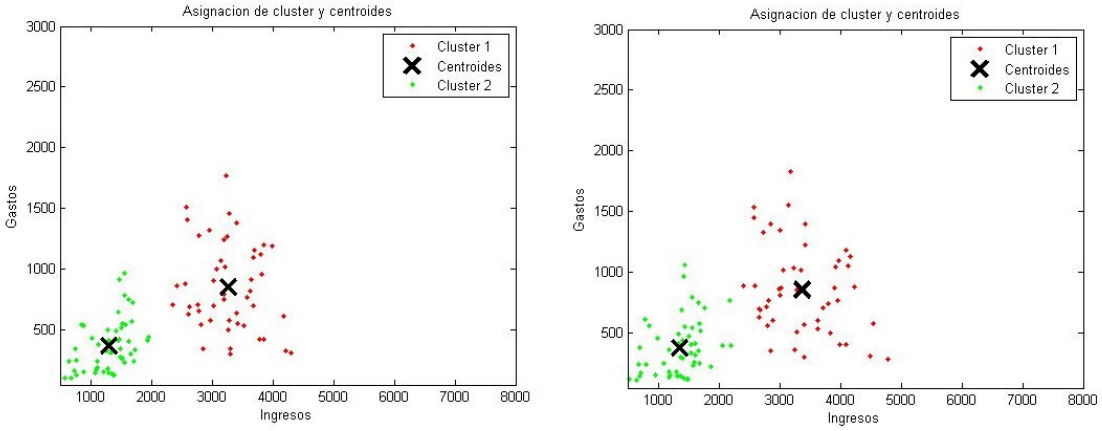

Figura 5. Asignación de cluster y centroides para una población dada para un periodo P=30 (Panel Izquierdo) y un P=45 (Panel Derecho).

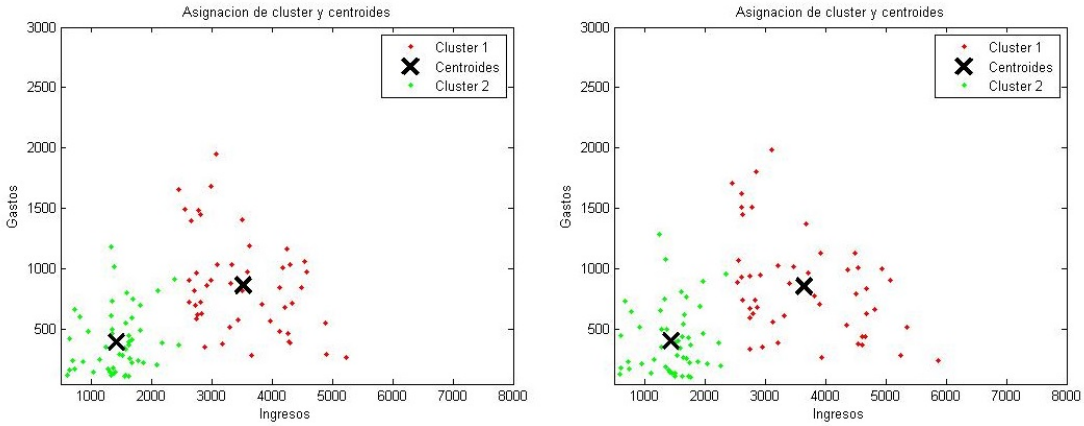

Figura 6. Asignación de cluster y centroides para una población dada para el periodo 60 (Panel Izquierdo) y el 75 (Panel Derecho).

En las figuras 4, 5 y 6 podemos visualizar las diferentes posiciones que ocupan los individuos y como cada cluster va cambiando en el tiempo, en un plano en el que los ingresos y los gastos se han representado en abscisas y ordenadas respectivamente.

Por consiguiente, y de forma paralela al desarrollo de los cluster a lo largo del tiempo se van creando unas estructuras que nos sirven de soporte para obtener los objetivos marcados. Estas estructuras, ya descritas anteriormente, tienen el fin de detectar tres cuestiones centrales.

En primer lugar el detectar las observaciones que se encuentran en la franja exterior del cluster. Desde el punto de vista del analista le puede ser muy interesante conocer los individuos que se encuentran más alejados del centroide del cluster. Para determinar qué observaciones son las que están más alejadas se utiliza la desviación estándar de cada observación, aunque su grado de desviación depende de la elección del analista. El principal interés de detectar esas observaciones alejadas puede servirnos para avisarnos de cuáles son las observaciones que debemos tener presentes como posibles candidatas a transitar a otro cluster. En la figura 7 tenemos una representación de los valores extremos para el ejemplo tipo en un instante de tiempo P=5.

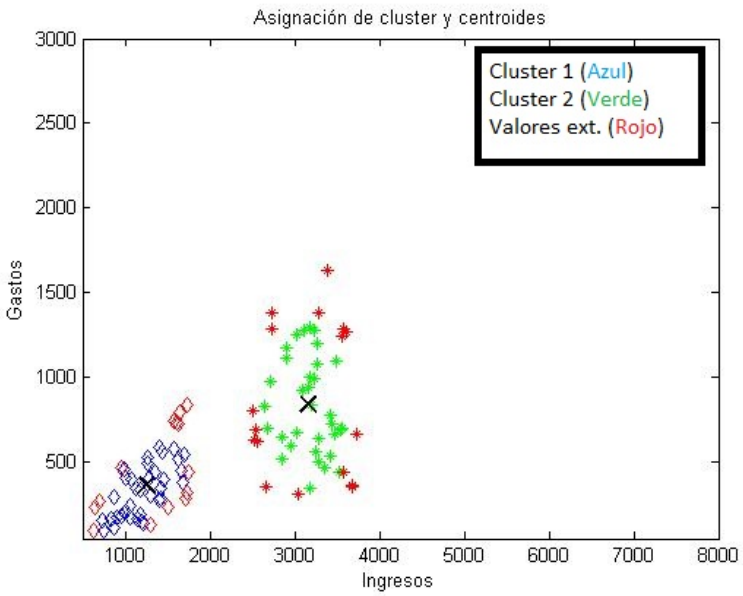

Figura 7. Representación de los valores extremos para cada cluster.

Como se puede apreciar en la figura, y sabiendo que los valores extremos son tomados mediante su desviación estándar dentro del clúster podríamos afirmar que: i) hay valores extremos que están más cerca del centroide, según una distancia euclídea, que otros valores que no se consideran extremos. Esto se debe a que se tiene en cuenta la estructura del clúster y por consiguiente los autovectores y autovalores que la conforman; ii) por otro lado, los valores que se encuentran en la franja exterior dan respuesta a uno de los objetivos marcados. Conocer estos valores aporta información para tomar decisiones con respecto a esta distinción.

En segundo lugar el algoritmo nos permite *detectar observaciones en peligro de transición*. Continuando con la idea anterior de observaciones que están cerca del límite del cluster, el concepto de cambio de cluster resulta aún más interesante. No todas las observaciones que se encuentran en una zona exterior del cluster tienen el mismo comportamiento o dinámica a lo largo del tiempo. La base para entender el concepto son las distancias de las observaciones a su centroide. Consideramos que una observación está en peligro de cambiar de un cluster a otro cuando su distancia hacia el cluster vecino es menor que la distancia hacia el centroide de su cluster. Esta distancia puede ser en la base canónica sin tener en cuenta la distribución de las observaciones dentro de cada cluster, o bien las distancias con respecto a la base nueva que se conforma en cada cluster. Usar un tipo de distancia u otra obtendremos resultados similares pero diferentes en la mayoría de los casos. La implicación que hay de trasfondo es el grado de

pertenencia a un cluster por parte de las observaciones. La interpretación de obtener estas observaciones en peligro de cambio puede ser esencial para el analista dado que puede tomar una decisión o estrategia concreta respecto a estas observaciones en peligro de cambio. Si tomásemos el instante 5 como está representado en la figura 7 veríamos que no existe valores en peligro de cambio. Esto implica que no todas las observaciones que se encuentran en el extremo presentan opciones de transito de cluster. Hay que recordar que el concepto de tránsito de clúster se basa en las distancias de las observaciones tomando referencia los centroides de los clúster. En la Figura 8 tenemos la representación en el periodo 70. En ella vemos que hay tres individuos pertenecientes al segundo cluster y que se encuentran en peligro de cambio. En este caso particular todos los individuos que están en peligro de cambio pertenecen al mismo cluster. Esta información se guarda en una estructura donde te indica qué individuo está en peligro de cambio y hacia qué cluster transitará, en nuestro ejemplo sólo hay una opción.

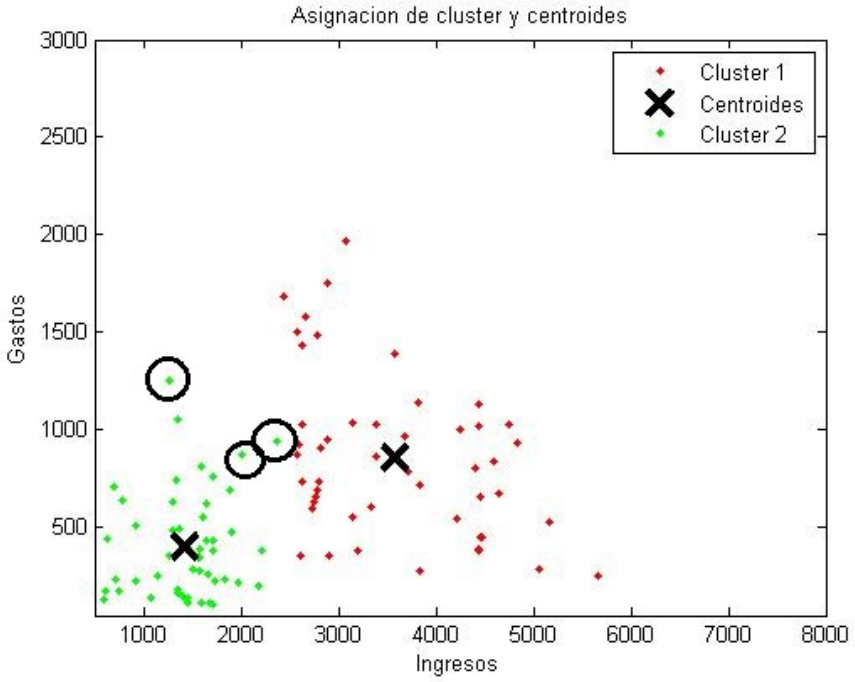

Figura 8. Representación de los valores en peligro de cambio en el periodo 70.

En tercer lugar, el enfoque proporciona una medida de *compacidad del cluster*. Otra de las medidas interesantes de estudio es la compacidad de un cluster tomada como el área que forman el producto de sus autovalores partido por el número de observaciones que la componen. Conocer la estructura que poseen los clusters implica poder dar un tratamiento diferente a cada uno de ellos. Podemos encontrarnos con clusters que presenten la misma cantidad de

observaciones pero debido a su compacidad diferente las conclusiones serán distintas. La compacidad también se puede interpretar desde el enfoque de las observaciones que forman el cluster. Clasificar los cluster según la dispersión de sus observaciones es un resultado muy interesante para este tipo de estudios.

6 Conclusiones y Extensiones

Este trabajo propuso un nuevo algoritmo que permite revisar una clasificación numérica, en base no a la geometría del espacio general de la nube de datos sino a la interna de cada grupo. Así, se trata de revisar una clusterización realizada con cualquier método de clusterización basado en distancias euclídeas sobre el conjunto de los datos, para tener en cuenta que algunas observaciones estarían potencialmente mejor clasificadas en otros grupos si tenemos en cuenta la geometría interna de cada grupo, esto es, si consideramos que ciertas variables son más importantes que otras a la hora de definir ese grupo. De esta forma el algoritmo que se propone reevalúa la clasificación inicial comprobando si, atendiendo a las componentes principales de cada grupo, hay observaciones que cambiarían de clúster.

La cuestión es especialmente importante en el contexto del análisis masivo de datos, ya que en cierta medida y potencialmente, la disponibilidad de un gran número de variables con importancia relativa distinta en aquello que pretendemos puede ser fuente de sesgo en la clasificación. Esta reevaluación atendiendo a los pesos introducidos por las componentes principales –es decir atendiendo a la geometría interna de cada grupo– puede ayudarnos a reclasificar observaciones probablemente mal clasificadas con la geometría general de la nube de puntos. La detección de las probabilidades de transitar entre cluster y la compacidad han sido dos de las aportaciones principales de este enfoque.

La aplicabilidad de este algoritmo a diferentes situaciones y contextos de cara a la segmentación, la estimación directa sin necesidad de la clasificación previa –aunque genere problemas computacionales– o la automatización de la detección de la probabilidad de transitar entre grupos son algunas de las extensiones a las que pretendemos dedicar nuestra investigación futura.

Referencias

Arthur, D. and Vassilvitskii, S. (2007). k-means++: The Advantages of Careful Seeding. *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. 1027-1035.

Bezdek, J.C., Ehrlich, R. & Full, W. (1984). FCM: The Fuzzy c-Means Clustering Algorithm. *Computers & Geosciences*, 10(2-3): 191-203.

de Amorim R.C. (2015). Feature Relevance in Ward's Hierarchical Clustering Using the Lp Norm. Journal of Classification, 32 (1), 46-62.

Defays, D. (1977). An efficient algorithm for a complete link method. *The Computer Journal. British Computer Society*, 20 (4), 364-366.

Dempster; P., Laird N.M., Rubin, D.B. (1977). Maximum Likelihood for Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 1-38.

Forgy, E.W. (1965). Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *Biometrics* 21, 768-769.

Hamerly, G. & Elkan, C. (2002). Alternatives to the k-means algorithm that find better clusterings. Proceedings of the eleventh international conference on Information and knowledge management (CIKM).

Kanungo, T., Mount, D.M., Netannyahu, N.S., Piatko, C.D., Silverman, R., & Wu, A.Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans. Pattern Analysis and Machine Intelligence* 24, 881-892.

Kaufman, L., & Roussew, P. J. (1990). *Finding Groups in Data - An Introduction to Cluster Analysis*. Wiley-Science Publication. John Wiley & Sons.

Li, B. (2006). A New Approach to Cluster Analysis: The Clustering-Function-Based Method. *Journal of the Royal Statistical Society*. Series B (Statistical Methodology, 68(3), 457-476.

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory,* 28 (2), 129-137

Maimon, O. & Rokach, L. (2005). Data Mining and Knowledge Discovery Handbook. Springer-Verlag.

Marx, V. (2013). The big challenges of big data. *Nature,* 498(7453), 255-260.

Ruspini, E.H. (1969). A New Approach to Clustering. *Information and Control* 15, 22-32.

Sibson, R. (1973). SLINK: an optimally efficient algorithm for the single-link cluster method. *The Computer Journal. British Computer Society.* 16 (1), 30-34.

Sokal, R. & Michener C. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38, 1409-1438.

Varian, H.R. (2013). Big Data: New Tricks for Econometrics. *Journal of Economic Perspectives*, 28 (2), 3-28.

Ward, J. H., Jr. (1963). Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association,* 58, 236-244.

Xu, D. And Tian, Y. (2015). A Comprehensive Survey of Clustering Algorithm. Annals of Data Science 2(2), 165-193.

Zadeh, L. (1965) Fuzzy sets. *Information and Control* 8, 338-353.

Zhang, W. E, Wang, X. Zhao, D. & Tang, X. (2012). Graph Degree Linkage: Agglomerative Clustering on a Directed Graph. 12th European Conference on Computer Vision, Florence, Italy, October 7–13, 2012.

Apéndice A: Códigos de análisis

```
%% Estructura de clusters

    clear,clc, close all;

    load poblacion_12

% Cargo la base de datos

    Poblacion = [Ingresos(:,1) Gastos(:,1)];

% Matriz con los datos; observaciones por filas y variables por columnas


    nClusters = 2;% numero de clusters que queremos agrupar

    NT = 100; % numero de individuos

    P = 84; % periodo temporal

%% Genero N clusters de mi poblaciÃ³n inicial con kmeans

CC = calculaParticionKmeans(Poblacion, nClusters,'dist','sqEuclidean');


%% Genero N clusters de las siguientes poblaciones

for i = 1:P


        Poblacion = [Ingresos(:,i) Gastos(:,i)];


        centros = [];

        for ii=1:length(CC)

            centros  = [centros ; CC(ii).centrosBC];

        end

CC = calculaParticionKmeans(Poblacion, nClusters,'dist','sqEuclidean','start',centros);


%% Representacion de los clusters


       for i = 1:length(CC)

           ptsymb = {'r.','g.','b.'};

           figure(20);
```

```
            plot(CC(i).obs(:,1),CC(i).obs(:,2),ptsymb{i},'MarkerSize',12)


            hold on
            plot(CC(i).centrosBC(:,1),CC(i).centrosBC(:,2),'kx',...
            'MarkerSize',15,'LineWidth',3)


            legend('Cluster 1','Centroides','Cluster 2','Location','NE')
            title 'Asignacion de cluster y centroides'
            axis ([500,8000,50,3000]);
            xlabel ('Ingresos');
            ylabel ('Gastos');


        end
        pause(0.2)
        hold off
        EC = estudioCluster(CC);


end
%% Representación de las observaciones que se encuentran en el extremo


for i = 1:P


         Poblacion = [Ingresos(:,i) Gastos(:,i)];


         centros = [];
         for ii=1:length(CC)
             centros  = [centros ; CC(ii).centrosBC];
         end


CC = calculaParticionKmeans(Poblacion, nClusters,'dist','sqEuclidean','start',centros);


for i= 1:length(CC)
```

```
        ptsymb = {'bd','g*'};
        figure(21);
        plot(CC(i).obs(:,1),CC(i).obs(:,2),ptsymb{i});
        hold on
        plot(CC(i).centrosBC(:,1),CC(i).centrosBC(:,2),'kx',...
           'MarkerSize',12,'LineWidth',2)
        for j = CC(i).obsExtrNb
            ptsymb = {'rd','r*'};
            plot(CC(i).obs(j,1),CC(i).obs(j,2), ptsymb{i});
            title ('Valores extremales');
            axis ([500,8000,50,3000]);
            xlabel ('Ingresos');
            ylabel ('Gastos');
        end
    end
       pause(0.2)
       hold off
end


function CC = calculaParticionKmeans(X,N,param1,value1,param2,value2)


%Funcion que realiza clusters sobre una poblacion
%   El output que obtenemos es una estructura que se repite para cada
%   cluster y que se compone de:
%   - CC(i).centrosBC = centros en la base canonica.
%   - CC(i).obs = coordenadas en la base canonica de los individuos.
%   - CC(i).eigVector = Autovectores.
%   - CC(i).eigValor = Autovalores.
%   - CC(i).nObs = numero de observaciones.
%   - CC(i).obsCentradas = observaciones centradas en funcion de su centro.
%   - CC(i).obsNb = observaciones en la nueva base.
%   - CC(i).obsDist = distancias de las observaciones a su centro.
```

```
%   - CC(i).obsDistNb = distancias de las obs a su centro en la nueva base.
%   - CC(i).compacidad = area del cluster/n° de observaciones.
%   - CC(i).obsExtr = observaciones que estan mas alejadas del centro
%   - CC(i).obsExtrNb = obs que estan mas alejadas del centro en la nueva
%   base.
    if nargin == 2
        [IDX,centros]=kmeans(X,N);
    elseif nargin == 4
        [IDX,centros]=kmeans(X,N,param1,value1);
    elseif nargin == 6
        [IDX,centros]=kmeans(X,N,param1,value1,param2,value2);
    end

    vd = 3; % valor de desviación estándar a considerar.

    for i = 1:N
        CC(i).centrosBC = centros(i,:);
        CC(i).obs = X(IDX==i,:);



        CC(i).nObs = length(CC(i).obs);

[pc,score,latent,tsquare] = princomp(CC(i).obs);
        CC(i).eigVector  = pc;
        CC(i).eigValor   = sqrt(latent);

        CC(i).compacidad = (CC(i).eigValor(1,1)*CC(i).eigValor(2,1))/CC(i).nObs;

        CC(i).obsCentradas = [CC(i).obs - ones(CC(i).nObs,1)*CC(i).centrosBC];
        Vpond = (pc*diag(CC(i).eigValor))';
        CC(i).obsNb = CC(i).obsCentradas / Vpond;
```

```
CC(i).obsDist =(sqrt(sum(CC(i).obsCentradas.^2')))';

        CC(i).obsExtr = find(CC(i).obsDist >= vd*std(CC(i).obsDist));


        CC(i).obsDistNb =(sqrt(sum(CC(i).obsNb.^2')))';

        CC(i).obsExtrNb = find(CC(i).obsDistNb >= vd*std(CC(i).obsDistNb));


    end


end

function EC = estudioCluster(CC)

%  input   CC.

%   Esta funcion trata de realizar un estudio para comprobar que

%   observaciones que pertenecen a un determinado cluster deberian

%   estar en otro cluster, debido a que sus distancias con el centro del

%   cluster vecino es menor que la distancia con el centro de su propio

%   cluster.


%   El output que obtenemos es una estructura que almacenamos en

%   - Obs_centradas12 = Observaciones del cluster1 centradas respecto al

%   centro del cluster 2.

%   - Obs_centradas21 = Observaciones del cluster2 centradas respecto al

%   centro del cluster 1.

%   - Obs_nuevas12 = Observaciones del cluster1 usando la base del

%   cluster2.

%   - Obs_nuevas21 = Observaciones del cluster2 usando la base del

%   cluster1.

%   - Distancia_12 = Distancia de cada observacion del cluster1 al

%   centro del cluster2.

%   - Distancia_21 = Distancia de cada observacion del cluster2 al

%   centro del cluster1.

%   - [C1] = Aqui se almacenan los indices de las observaciones que

%   perteneciendo al cluster 1, sus distancias son menores al cluster2.
```

```
%   - [C2] = Aqui se almacenan los indices de las observaciones que
%   perteneciendo al cluster 2, sus distancias son menores al cluster1.
%   - EC = Aqui se guarda los valores en peligro de cambio indicando en la
%   primera columna el cluster al que pertenece, en la segunda columna al
%   cluster al que transitaria y en la tercera columna el indice del
%   individuo.

EC =[];

for j = 1:(length(CC)-1)
    for i = j+1:length(CC)
        if j~=i
            C1 = CC(j);
            C2 = CC(i);

            Obs_centradas12 = [C1.obs - ones(C1.nObs,1)*C2.centrosBC];
            Obs_centradas21 = [C2.obs - ones(C2.nObs,1)*C1.centrosBC];

            Obs_nuevas12 = Obs_centradas12 / (C2.eigVector*diag(C2.eigValor))';
            Obs_nuevas21 = Obs_centradas21 / (C1.eigVector*diag(C1.eigValor))';

            Distancia_12 =(sqrt(sum(Obs_nuevas12.^2')))';
            Distancia_21 =(sqrt(sum(Obs_nuevas21.^2')))';

            aux = find(Distancia_12<C1.obsDistNb);
            EC = [EC ; [ones(size(aux))*j,ones(size(aux))*i, aux]];

            aux = find(Distancia_21<C2.obsDistNb);
            EC =[EC ;[ones(size(aux))*i, ones(size(aux))*j, aux]];
        end
    end
end
```

Apéndice B: Códigos de población

```
%% Fichero principal  main.m

clc, close all, clear all;


%% Definimos los parametros de la sesion


NT = 100;        % Numero de individuos

P  = 84;        % Longitud del intervalo temporal

Base = 400;      % valor de origen de la serie tendencial

p2  = 0.02;       % Probabilidad de cambio en los factores a largo plazo


% Parametros

alfa = 1/800;

beta = 1/800;


%% Definimos la tendencia "tend" y la dibujo

tend = tendencia(P,Base);


plot( 1:P, tend);

title('tendencia economica');

xlabel('periodo en meses');

grid on


%% Generacion de los factores a largo plazo


F_lp = factor_lp(P,NT,p2);


%% Generacion de los factores a corto plazo


F_cp = factor_cp(P,NT);
```

```
%% Generacion de los datos de ingresos para el periodo y los individuos
%   Ingresos en funcion de una tend, factores a largo y corto plazo

Ingresos = ingresos(NT,P,tend,F_lp,F_cp,alfa,beta);

%% Generacion de los datos de gastos para el periodo y los individuos

Gastos = gastos(Ingresos,NT,P,tend,F_lp,F_cp,alfa, beta);

%% Representacion de los datos en el tiempo para los individuos

% FCP = factores a corto plazo representando un individuo en un periodo P
figure(1);
plot(F_cp(1,1:P),'-bo','LineWidth',1);
xlabel('Periodo en meses');
title('Factores a corto plazo para un individuo n');

minI = min(Ingresos(:));
maxI = max(Ingresos(:));
minG = min(Gastos(:));
maxG = max(Gastos(:));

% Representación de los ingresos y gastos de 4 individuos a lo largo del
% tiempo.
figure(2);
subplot(2,2,1);plot(Ingresos(24,:),'b-');
hold on
plot (Gastos(24,:),'r-');
xlabel('periodo en meses');
ylabel('cantidades en euros');
subplot(2,2,2);plot(Ingresos(33,:),'b-');
```

```
hold on
plot (Gastos(33,:),'r-');
xlabel('periodo en meses');
ylabel('cantidades en euros');
subplot(2,2,3);plot(Ingresos(68,:),'b-');
hold on
plot (Gastos(68,:),'r-');
xlabel('periodo en meses');
ylabel('cantidades en euros');
subplot(2,2,4);plot(Ingresos(87,:),'b-');
hold on
plot (Gastos(87,:),'r-');
xlabel('periodo en meses');
ylabel('cantidades en euros');
hold off

% Representacion de los individuos a lo largo del tiempo
   figure(3);
   axis([minI maxI minG maxG]);
for i = 1:P
   plot(Ingresos(:,i),Gastos(:,i),'r.');
   pause(0.3);
end
save 'poblacion_13' 'Ingresos' 'Gastos'
function [ Y ] = tendencia(X, Base)
% Esta funcion genera una dinamica segun unos valores de X. La funcion
% simula la tendencia economica global de un pais

% ------ SALIDAS -------
    % Y = la dinamica economica de un pais

% ------ ENTRADAS ------
```

```
    % X = periodo en el que se proyecta la dinamica
    % Base = el valor de origen de Y

    XX = 1:X;

    Y = XX/2 + 3/2*sin(2*pi*XX/12)+Base;

end

function FLP = factor_lp(P,NT,p2)
% Factores a largo plazo
% Los factores a largo plazo son aquellas variables, decisiones propias de
% cada individuo y que cambian muy poco en el tiempo. Un ejemplo de esto es
% el nivel educativo, el lugar donde reside, familiares dependientes...etc.
% En este codigo, esos factores son clasificados en una escala de -5 a 5,
% donde cuanto mas cercano a 5, estos individuos tienden a alcanzar su
% mayor capacidad o rendimiento dentro de sus posibilidades. Si su factor
% es negativo el efecto es contrario.  Esto se refleja en los ingresos y
% gastos de cada individuo.

% --------- SALIDAS ----------
% FLP = Factor a Largo Plazo; matriz donde se guarda los valores aleatorios

% --------- ENTRADAS ---------
    %  P = periodos
    %  NT = n° de individuos
    %  p2 = umbral de cambio

% Genero un vector inicial con valores aleatorios para T individuos entre -5 y 5.
    FLP = round(-5 + (5+5)*(rand(NT,1)));
```

```
% Matriz inicial de probabilidades entre 0 y 1
    p1 = rand(NT,P);

% En este bucle se rellena la matriz FLP segun las probabilidades de
    % cambiar de factor. Para que se produczca el cambio la probabilidad tiene
    % que ser menor del umbral p2. En ese caso se estima el grado de cambio
    % que puede tomar valores entre -2 y 2.
    for i=1:NT
        for j=1:P-1
            if p2 >= p1(i,j);
                salto = round(-4*rand+2);
                FLP(i,j+1)= FLP(i,j) + salto;

                % Cotas superiores e inferiores -5 y 5
                FLP(i,j+1) = max(-5,min(5,FLP(i,j+1)));
else
                FLP(i,j+1)= FLP(i,j);
            end
        end
    end
end

function FCP = factor_cp(P,NT)
% Los factores a corto plazo son aquellas decisiones diarias que un
% individuo toma a lo largo de un periodo. Estas decisiones producen
% cambios debiles en la dinamica general de ingresos y gastos de un individuo.
% Este factor sera aleatorio en un intervalo del 5% positivo o negativo
% sobre los ingresos y gastos en el momento actual.
% ---------- SALIDA ---------------
    % FCP = Factor a Corto Plazo; es la matriz donde guardamos los valores
    % aleatorios.
% ---------- ENTRADA --------------
```

```
    % P = periodo temporal
    % NT = n° de individuos


    FCP = (-5 + (5+5)*rand(NT,P));


end


function ING = ingresos(NT,P,tend,F_lp,F_cp,alfa,beta)
% La funcion ingresos genera una dinamica de los ingresos de un numero de
% individuos en un periodo dado. Para ello se tiene en cuenta la tendencia
% economica global y las modificaciones en los ingresos en el largo y corto
% plazo.

% ------- SALIDAS ----------
    % ING = Matriz donde se recoge los ingresos de cada individuo en un periodo
    % determinado.

% ------- ENTRADAS --------
    % NT = numero total de individuos
    % P = periodo temporal
    % tend = tendencia global
    % F_lp = matriz de los factores a largo plazo
    % F_cp = matriz de los factores a corto plazo
    % alfa = parametro de ajuste de los factores a largo plazo
    % beta = parametro de ajuste de los factores a corto plazo

    % Genero dos muestras de diferentes niveles de ingresos iniciales (ING1 y  ING2)
    n = round(NT/2);
```

```
    ING1 = (600+(1800-600).*rand(n,1));

    ING2 = (2500+(3700-2500).*rand((NT-n),1));


%Conformo una matriz unica llamada ING

    ING = [ING1;ING2];


% En este bucle se genera la dinamica de los ingresos teniendo en

    % cuenta los ingresos del instante anterior, la variacion de la

    % tendencia entre el instante actual y el anterior y las modificaciones

    % debidas al largo y corto plazo.

    for i = 2:P

        ING(1:NT,i) = ING(1:NT,i-1) + ...     % instante anterior

                          ((tend(i)- tend(i-1))/tend(i-1)) * ING(1:NT,i-1)+ ... % incremento por
la tendencia

                          (ING(1:NT,i-1) .* F_lp(1:NT,i-1)) * alfa + ...   % Modificaciones
debidas al largo plazo

                          ING(1:NT,i-1) .* F_cp(1:NT,i-1) * beta ;    % Modificaciones debidas
al corto plazo

    end


end


function GAST = gastos (Ingresos,NT,P,tend,F_lp,F_cp,alfa, beta)


%% Generacion de los datos de gastos para el periodo y los individuos


% Vector inicial llamado gastos para cada individuo. debe estar

% comprendido entre el 40 y el 80 % de los ingresos.


GAST = [Ingresos(:,1)].*((40+(80-40).*rand(NT,1))./100);


    for i = 2:P

        GAST(1:NT,i) = GAST(1:NT,i-1) + ...     % instante anterior
```

```
                        ((tend(i)- tend(i-1))/tend(i-1)) * GAST(1:NT,i-1) - ... % incremento
por la tendencia

                        (GAST(1:NT,i-1) .* F_lp(1:NT,i-1)) * alfa - ...   % Modificaciones
debidas al largo plazo

                        GAST(1:NT,i-1) .* F_cp(1:NT,i-1) * beta ;    % Modificaciones debidas
al corto plazo

    end

end
```

Agradecimientos

El autor quiere agradecer al profesor M.E. Gegúndez por su infinita paciencia y por todo lo que me ha enseñado a lo largo de estos meses. Los méritos de este trabajo son, sin duda, suyos. Igualmente, me gustaría agradecer a los compañeros y al cuadro de profesores del programa su ayuda y el haber contribuido a establecer un clima de trabajo que ha favorecido el desarrollo de todos.