Clustering de series de tiempo con datos categóricos

Autor

Javier Rivera Pinto

Tutores

José Antonio Vilar Fernández
Manuel García Magariños

Mediante el presente escrito, José A. Vilar Fernández y Manuel García Magariños, hacemos constar lo que sigue.

1. Que el alumno del Máster en técnicas estadísticas D. Javier Rivera Pinto ha realizado el trabajo titulado *Clustering de series de tiempo con datos categóricos*, en el que figuramos como directores.

2. Que la memoria que se acompaña constituye la documentación que, con nuestra autorización, el alumno entrega al objeto de presentar y defender su trabajo como Proyecto Fin de Máster.

En A Coruña, a 30 de junio de 2014

Firmado: José A. Vilar Fernández

Firmado: Manuel García Magariños

Resumen

El análisis cluster es una herramienta estadística basada en fragmentar un conjunto de objetos en grupos homogéneos o clusters. Su implementación con datos multivariantes (Cluster Analysis, 5th Edition, Brian S. Everitt, Sabine Landau, Morven Leese, Daniel Stahl) o series de tiempo (Pattern Recognition Clustering of TS Data A Survey, Liao 2005) es largamente conocida. Por otra parte, las series de tiempo con respuesta nominal son cada vez más comunes en la era de las tecnologías de la información: secuencias de ADN, navegación de usarios en Internet, etc. No obstante, el desenvolvimiento de algoritmos cluster para ser aplicados a este tipo de datos ha recibido escasa atención en la literatura científica. En una primera parte, este trabajo presenta una revisión de metodologías cluster existentes, incluyendo procedimientos basados en modelos previos y en el cómputo de disimilaridades entre pares de objetos. Posteriormente, la conducta de estas técnicas se examina sobre bases de datos simulados, imitando escenarios posibles para secuencias de datos nominales. La comparación de los resultados permite extraer conclusiones sobre los contextos apropiados de aplicación de cada técnica.

Agradecimientos

Me gustaría dar gracias a toda aquella persona que haya leído, esté leyendo o vaya a leer este documento. A quienes me han educado desde pequeño poniéndo todo de su parte para que pueda lograr lo que quiero, a quienes han compartido los mejores y peores momentos de mi vida, a quienes me han formado en un mundo complejo pero real como es el de las matemáticas y a quienes me han ayudado en una corrección, redacción y organización adecuada a las exigencias . . . ¡gracias por hacer posible que haya escrito este trabajo!

Por otra parte, agradecer también el tiempo empleado a quien vaya a leer lo que a continuación sigue; esperando que le pueda resultar de utilidad.

Índice general

Introducción

El análisis cluster está compuesto por procedimientos estadísticos cuyo fin es el de clasificar a un conjunto de datos en varios grupos o clusters, en función de los valores que tomen las variables que los definen. Esta clasificación se realiza sin conocimiento previo de cuáles son ni cuántos grupos existen (salvo que se trabaje con datos simulados). Por este motivo y ante el desconocimiento a priori de la asignación correcta de los individuos a los grupos, el análisis cluster forma parte las denominadas técnicas de aprendizaje no supervisado (*unsupervised learning techniques*).
Mediante las diversas técnicas estadísticas, los conjuntos que se forman deben venir definidos de tal manera que los elementos que se encuentren en el mismo grupo presenten características comunes y sean lo más homogéneos posibles. Además, para una buena clasificación resulta conveniente que los distintos clusters propuestos asuman características que los diferencien, obteniendo así heterogeneidad entre los grupos.

Formar los conglomerados no es una tarea fácil ante el desconocimiento de los patrones que los rigen, del número de ellos que existen y de algo que resulta más trascendental: saber si verdaderamente tiene sentido considerar clusters en base a la naturaleza de los individuos. El conocimiento por parte del usuario acerca de la base de datos con la que se está trabajando es un punto importante a tener en cuenta. No existe un algoritmo universal para el análisis cluster que permita clasificar en grupos un conjunto de datos cualquiera. Resulta crucial conocer el origen de las observaciones, puesto que saber con qué tipo de datos se trabaja y tener cierta intuición a cerca de lo que se puede (o quiere) obtener, permite concretar algunos aspectos de interés que facilitarán la resolución del problema. Además, será posible determinar los métodos más adecuados para la estimación de algunos de los parámetros que requerirán los modelos presentados para la labor.

La necesidad de determinar grupos homogéneos de datos se presenta en muchos ámbitos: distintos grupos de usuarios en una web, de personas con síntomas parecidos, de empresas con características similares, ... Es por ello que las técnicas de clustering resultan de gran ayuda en diversos campos tales como: minería de datos, aprendizaje de máquinas, genética, medicina, economía, bioinformática, ... Ante tanto interés por parte de múltiples áreas de conocimiento, el análisis cluster es un tema que ha sido ampliamente tratado por muchos científicos. A pesar de ello, existen campos en los cuáles no se ha profundizado del todo debido a la gran variedad de datos con la que se puede trabajar: datos

numéricos, temporales, categóricos, gráficos, ... La mayoría de algoritmos existentes en la literatura son referentes a datos con variables numéricas: k - medias, k - modas, algoritmos jerárquicos, ... Con lo que a datos categóricos respecta, es sensiblemente menor el número de aportaciones. Uno de los motivos por los que puede ser así, es la naturalidad con la que se pueden definir distancias entre datos numéricos y la dificultad que implica hacerlo con datos categóricos.
Otro factor a tener en cuenta y que concierne a uno de los ejemplos tratados a lo largo del presente trabajo, es el dinamismo que pueden presentar los individuos y sus observaciones.
Cuando las variables tratadas en el análisis presentan una evolución a lo largo del tiempo, las técnicas de clasificación consideradas deben tener en cuenta el carácter ordenado de las secuencias de los registros. Existen diferentes vías para captar esa característica de los registros, por ejemplo considerar modelos de series temporales subyacentes a los datos que tengan en cuenta el carácter dinámico de los mismos, o definir criterios de distancia o disimilaridad entre observaciones que igualmente contemplen afinidad entre estructuras de dependencia temporal.

Al igual que ocurre en el caso de datos sin dependencia temporal subyacente, el análisis cluster de series temporales con respuesta continua ha sido ampliamente abordado en la literatura. Dos excelentes revisiones de los avances en este campo son Liao (2005) y, más recientemente y en el área de la inteligencia artificial y la minería de datos, Fu (2011). Pese a no existir una variedad extensa de métodos para el clustering de series con variables categóricas, diversas técnicas propuestas para ello han permitido ir construyendo modelos cada vez más complejos y que van recogiendo más características de la estructura de los datos tales como Huang (1998) o Ahmad and Dey (2007).

Los dos grandes tipos de algoritmos que se encuentran en la literatura referentes a análisis cluster de datos con carácter numérico son: los basados en un modelo y los basados en disimilaridad.
Mientras los algoritmos basados en el modelo asumen una estructura dinámica del conjunto de datos modelizada mediante cadenas de Markov o mixturas de éstas, los algoritmos basados en disimilaridad utilizan una distancia o métrica entre individuos para poder clasificarlos. Si bien una extensión de los primeros a variables con respuesta categórica es factible (Pamminger and Frühwirth-Schnatter (2010) o Cadez et al. (2003)), con los basados en una distancia o métrica surgen dificultades. Ya de por sí es complicado definir una distancia entre variables categóricas; con lo que mayor es la dificultad de obterner una métrica entre secuencias ordenadas de datos categóricos que contemple posibles dependencias temporales.

El presente trabajo se enmarca por tanto en el contexto del análisis cluster de secuencias temporales de registros categóricos. Se plantean dos objetivos, por un lado proporcionar una revisión general de los diferentes caminos seguidos en la literatura para abordar este problema. De entrada se considera incluso el enfoque tradicional de clustering para datos categóricos no dependientes, con objeto de

mostrar su ineficacia en el contexto considerado y justificar así el estudio de vías alternativas de análisis. Se presentan entonces algunas de estas técnicas, todas ellas muy recientes. El segundo objetivo es proceder a la evaluación y comparación del comportamiento de algunos de estos procedimientos. Para ello se simulan diferentes escenarios posibles y se ejecutan analizando a continuación la calidad de las soluciones clustering experimentales. Las pruebas realizadas permiten obtener conclusiones sobre las ventajas y desventajas de todos ellos así como de los escenarios de aplicación apropiados. Para generar las bases de datos experimentales y desarrollar el software se ha utilizado el lenguaje estadístico R (http://www.r-project.org/). Algunos de los algoritmos utilzados en las simulaciones se encuentran en paquetes de libre disposición dentro del programa, si bien otros han sido cedidos por los autores. En ocasiones ha sido necesaria la modificación puntual de los mismos a fin de adaptarlos a las situaciones propuestas.

Por último, a la vez que se comentan las ventajas e inconvenientes de los algoritmos empleados, se dan indicaciones de posibles campos de trabajo en un futuro, relacionados con este tema de investigación.

Capítulo 1

Revisión de procedimientos de análisis cluster para datos con respuesta categórica

A continuación se presentan algunas de las metodologías publicadas en lo referente al *análisis cluster* de datos con respuesta categórica. Se parte de los procedimientos más simples (aquellos que no contemplan dependencia temporal en los registros) y se van construyendo algoritmos más complejos. A diferencia de las propuestas iniciales, las más recientes recogen propiedades de los datos tales como correlación entre individuos y dependencia temporal entre las variables.

1.1. Sin considerar el carácter dinámico

En referencia a algoritmos que no tienen en cuenta el carácter temporal o dinámico sobre la sucesión de respuestas categóricas que define al individuo, se encuentra Huang (1998). Resulta de relevancia pues describe una de las primeras propuestas para la realización de análisis cluster con este tipo de datos. Huang (1998) hace referencia a técnicas propuestas previamente por otros autores y justifica la falta de utilidad de las mismas. Posteriormente, toma como base el algoritmo de las k - medias para dar lugar a una extensión de este procedimiento para secuencias de valores categóricos. Surge así el algoritmo de las k - modas y, fruto de la naturaleza mixta de los datos (con presencia de variables categóricas y variables numéricas), concluye con la presentación de una técnica para los mismos denominada k - prototipos.

Referente a las técnicas desarrolladas hasta ese momento, cabe destacar que dos de los procedimientos estándar para el análisis cluster más frecuentes son los métodos jerárquicos y los métodos partitivos.

Los algoritmos jerárquicos son algoritmos secuenciales en los que cada etapa consiste en unir o separar grupos. Permiten la clasificación de datos con los dos tipos de variables (numéricas y categóricas) si bien el coste computacional no los convierte en métodos aceptables para el análisis cluster cuando se trabaja con datos de altas dimensiones. Los algoritmos partitivos constan desde el inicio con un número fijo de clusters y son los individuos los que se van moviendo de uno a otro en base a un criterio determinado, hasta llegar a un punto de estabilidad. El algoritmo de las *k*-medias (explicado más adelante) es uno de los algoritmos partitivos más conocidos y resulta eficiente de cara a analizar conjuntos de datos extensos, si bien sólo es válido para datos numéricos.
Un intento de numerizar los datos categóricos puede resultar útil aunque, en la mayoría de los casos este proceso conlleva a la pérdida de significación de los resultados ofrecidos por el algoritmo. Por esa vía, Ralambondrainy por ejemplo presenta en Ralambondrainy (1995) una propuesta para utilizar el algoritmo de las *k*-medias en el clustering de datos categóricos. En ella, cada categoría de cada variable se representa con un número binario. En cada posición, el número presenta un 1 en caso de que el individuo en cuestión posea la característica y un 0 en caso de ausencia. Las desventajas de este algoritmo vienen determinadas por el alto coste computacional que puede requerir (en el caso de un número alto de posibles respuestas categóricas por variable) y por la falta de significación de los resultados obtenidos al aplicar el algoritmo de las *k*-medias.

Por tanto, todos los métodos existentes hasta entonces basados en métodos para variables numéricas y los correspondientes algoritmos; presentaban deficiencias en cuanto a la interpretación de los resultados si con variables categóricas se trabajaba. Con el propósito de solventar el problema este que presenta el análisis cluster con datos categóricos, en Huang (1998) se proponen dos algoritmos inspirados en el de las *k*-medias pero aplicables a datos de esta naturaleza. Estas propuestas son: el algoritmo de las *k*-modas y el algoritmo *k*-prototipos. Previo al desarrollo de los algoritmos, conviene introducir la notación que se emplea en los mismos.

- El conjunto global de datos se denota por $\mathbf{X}$ y viene determinado por cada uno de sus n elementos $\mathbf{X} = \{X_1, \ldots, X_n\}$
- Cada uno de los individuos del conjunto $\mathbf{X}$ viene determinado por el valor específico de las m variables aleatorias $A_1, \ldots, A_m$.
  Todos los individuos que forman el conjunto de datos presentan un valor concreto para cada una de las variables.
- Las variables aleatorias $A_i \ : \ i \in \{1 \ , \ldots , \ m\}$ toman valores dentro de su propio dominio $dom(A_i)$.
  Los dominios pueden estar compuestos por valores numéricos o categóricos. Así:

- si la variable es categórica, $dom(A_i)$ es finito y no presenta una relación de orden entre sus elementos.
- si la variable es numérica, $dom(A_i)$ presenta una relación de orden definida sobre todos sus elementos.

Independientemente del tipo de variable, ε es un elemento incluido en todos los dominios. Su presencia denota la ausencia de un valor para dicha característica.

- No existe ningún tipo de relación de inclusión entre las variables A_i a fin de que a la hora de definir distancias entre elementos del conjunto de datos no se generen problemas.
- La representación de un elemento del conjunto de datos, es el resultado de concatenar cada uno de los valores que toma cada una de las variables del análisis. Así, un individuo viene determinado por una de las siguientes expresiones:

$$[A_1 = x_1] \ \wedge \ [A_2 = x_2] \ \wedge \ \ldots \ \wedge \ [A_m = x_m] \ : \ x_j \in dom(A_j) \ \ \forall j \in \{1, \ldots, m\}$$

ó

$$[x_1^r \ , \ldots, x_p^r, x_{p+1}^c, \ldots, x_m^c]$$

siendo los p primeros elementos valores numéricos y el resto valores categóricos.

El hecho de que $X_i = X_j$ no determina que se trate del mismo individuo, sino que ambos comparten el mismo valor para todas y cada una de las variables aleatorias.

Como los algoritmos propuestos por Huang (Huang (1998)) están basados en el de las k - medias, resulta útil aclarar en qué consiste el método:

Algoritmo de las *k*-medias

El algoritmo de las k-medias es uno de los métodos de clustering más utilizados. Forma parte de los algoritmos de partición y fue diseñado para individuos definidos mediante variables numéricas con respuesta en intervalos contínuos. Su estructura es la siguiente:

- **Elementos de entrada**:
  - Conjunto de datos $\mathbf{X} = \{X_1, \ldots, X_n\}$
  - Número entero indicando la cantidad de conglomerados propuesta k ($k \leq n$)

- **Elementos de salida**:
  - Una partición de **X** en *k clusters* de forma que minimiza la suma de errores al cuadrado (distancias) dentro de cada grupo.

El planteamiento puede reescribirse como la resolución de un problema de programación de la siguiente forma:

$$\text{Minimizar} \quad P(W,\mathscr{C}) = \sum_{l=1}^{k}\sum_{i=1}^{n} w_{i,l}\, d(X_i, C_l)$$

sujeto a

$$\sum_{l=1}^{k} w_{i,l} = 1 \quad 1 \le i \le n$$
$$w_{i,l} \in \{0,1\} \;\; 1 \le i \le n,\; 1 \le l \le k$$

donde :

- W es una matriz de dimensión n x k.
- $\mathscr{C} = \{C_1, \ldots, C_k\}$ es un conjunto de elementos denominados centroides, pertenecientes al dominio del conjunto **X**. Se entiende por centroide un elemento representativo del cluster dado por $C_l = \{c_{l1}, \ldots, c_{lm}\}$.
- $d(\cdot,\cdot)$ representa la distancia euclídea entre dos objetos.

La resolución de este problema puede efectuarse de forma iterativa solventando los dos subproblemas siguientes:

1) Fijar $\mathscr{C} = \hat{\mathscr{C}}$ y resolver el problema $P(W, \hat{\mathscr{C}})$.

2) Fijar $W = \hat{W}$ y resolver el problema $P(\hat{W}, \mathscr{C})$.

Una forma de proceder con el primer subproblema es asignar un peso ($w_{i,l}$) con valor 1 al índice del cluster l con respecto del cuál el elemento X_i esté más cerca y considerar 0 el resto de pesos ($w_{i,j} \;:\; j \neq l$).

$$w_{i,l} = 1 \quad \text{si} \quad d(X_i, C_l) \le d(X_i, C_t) \quad 1 \le t \le k$$
$$w_{i,l} = 0 \quad \text{si} \quad t \neq l.$$

Para el segundo subproblema basta con determinar las componentes de cada uno de los centroides como la media de los individuos que forman el cluster.

$$c_{l,j} = \frac{\sum_{i=1}^{n} w_{i,l}\, x_{i,j}}{\sum_{i=1}^{n} w_{i,l}} \quad \text{para} \quad 1 \leq l \leq k \quad 1 \leq j \leq m.$$

Con ayuda de la resolución de los dos subproblemas, el algoritmo para resolver el planteamiento inicial queda como sigue:

1) Elegir un $\mathscr{C}^0$ inicial y resolver $P(W, \mathscr{C}^0)$ (subproblema 1) para obtener así W^0.

2) Considerar $\hat{W} = W^t$ y resolver $P(\hat{W}, \mathscr{C})$ para obtener $\mathscr{C}^{t+1}$ (subproblema 2).

- Si $P(\hat{W}, \mathscr{C}^t) = P(\hat{W}, \mathscr{C}^{t+1})$ finalizar (devolver $(\hat{W}, \mathscr{C}^t)$).
- Si $P(\hat{W}, \mathscr{C}^t) \neq P(\hat{W}, \mathscr{C}^{t+1})$ ir al paso 3.

3) Considerar $\hat{\mathscr{C}} = \mathscr{C}^{t+1}$ y resolver $P(W, \hat{\mathscr{C}})$ para obtener W^{t+1} (subproblema 1).

- Si $P(W^t, \hat{\mathscr{C}}) = P(W^{t+1}, \hat{\mathscr{C}})$ finalizar (devolver $(W^t, \hat{\mathscr{C}})$).
- Si $P(W^t, \hat{\mathscr{C}}) \neq P(W^{t+1}, \hat{\mathscr{C}})$ tomar $t = t + 1$ e ir al paso 2.

El coste computacional de este algoritmo es del orden $O(Tkn)$ siendo T el número de iteraciones, n el número de elementos del conjunto de datos y k el número de clusters.

Algunas de las propiedades que presenta este algoritmo son eficiencia para conjuntos de datos grandes, convergencia casi siempre tras varias iteraciones, utilidad exclusiva para datos con variables aleatorias numéricas (debido a la necesidad de llevar a cabo operaciones aritméticas que no pueden ser ejecutadas con otro tipo de datos) y convexidad de clusters que devuelve el algoritmo tras su ejecución.
Una misma ejecución del algoritmo de las k-medias, puede dar lugar a resultados distintos en función de la selección de los centroides iniciales, la definición de distancia (variantes a la euclídea) y el proceso de actualización del centroide de cada cluster.
Uno de los inconvenientes del algoritmo de las k - medias es la inoperabilidad del mismo ante datos formados por variables con respuesta categórica. En busca de un método similar que permita operar con este tipo de datos, Huang propone el siguiente algoritmo.

Algoritmo de las k - modas

Como se ha citado previamente, los problemas que surgen con variables categóricas son principalmente la ausencia de una métrica entre datos y la inexistencia de una media para los mismos.
La idea de este algoritmo se puede resumir en el hecho de que ante la falta de media, con datos categóricos se puede trabajar con la moda de una variable. Con respecto a la distancia entre dos individuos, se considera una métrica sencilla denominada SMD (*Simple Matching Dissimilarity*) que se presenta a continuación. El procedimiento consta de los siguientes pasos.

- Considerar la SMD como medida de distancia para datos categóricos.
- Sustituir las medias por modas en el algoritmo existente para variables numéricas.
- Emplear un metodo basado en frecuencias para hallar las modas y resolver la dificultad presentada en el segundo subproblema.

Paso I: cambio de medida

La medida para datos categóricos viene dada como sigue. Sean $X = [x_1, \ldots, x_m]$ e $Y = [y_1, \ldots, y_m]$ dos individuos del conjunto de datos con el que se desea trabajar (representados mediante m variables categóricas), se define la distancia d_1 entre ambos como:

$$d_1(X,Y) = \sum_{j=1}^{m} \delta(x_j, y_j)$$

siendo

$$\delta(x_j, y_j) = \begin{cases} 0 & \text{si } x_i = x_j, \\ 1 & \text{si } x_i \neq x_j. \end{cases}$$

La distancia considerada en las simulaciones ha sido la definida por Huang dividida entre el número de variables m a fin de obtener un valor en el intervalo $[0, 1]$

Paso II: sustitución de la media por la moda

Se define la moda de un conjunto $\mathbf{X}$ compuesto por elementos definidos mediante variables categóricas ($A_1, \ldots, A_m$) como el vector $C = [c_1, \ldots, c_m]$ que minimiza

$$D(\mathbf{X}, C) = \sum_{i=1}^{n} d_1(X_i, C).$$

Cabe la posibilidad de que la moda del conjunto **X** pueda resultar un vector C que no pertenezca a dicho conglomerado.

Paso III: método basado en frecuencias

Huang demuestra un resultado en Huang, 1998, pg. 302, que establece la asignación del centroide de cada grupo como la concatenación de los valores de cada una de las variables con mayor frecuencia relativa, para así minimizar la función objetivo $D(\mathbf{X},\mathscr{C})$. No obstante, el método definido por el teorema para hallar $\mathscr{C}$ dado un conjunto **X** da lugar a la posibilidad de que exista más de un resultado para éste.
Dicho teorema, entendiendo por $a_{i,j}$ la respuesta i-ésima que puede tomar la variable A_j viene dado por el siguiente enunciado:

La función $D(\mathbf{X},\mathscr{C})$ se minimiza si y sólo si $f_r(A_j = c_j \mid \mathbf{X}) \geq f_r(A_j = a_{i,j} \mid \mathbf{X})$ para $c_j \neq a_{i,j}$ $\forall\ j\ \in\ 1,\ldots,m$.

Teniendo en cuenta la distancia definida para datos con variables categóricas, la función objetivo $P(W,\mathscr{C})$ puede reescribirse como:

$$P(W,\mathscr{C}) = \sum_{l=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,l}\delta(x_{i,j},c_{l,j})$$

siendo $w_{i,l} \in W$ y $C_l = \left[c_{l,1}\ ,\ \ldots\ ,\ c_{l,m}\right]\ \in\ \mathscr{C}\quad 1\ \leq\ l\ \leq k$.

Una vez definidos todos los elementos, el algoritmo de las k-modas puede aplicarse siguiendo los siguientes pasos:

1) Seleccionar k centroides, una para cada cluster.

2) Ubicar cada individuo en el cluster cuyo centroide esté más cerca de acuerdo a la medida establecida. Inmediatamente después de la ubicación de un elemento en un cluster actualizar el centroide del mismo.

3) Tras asignar cada uno de los objetos a un cluster, volver a medir las distancias entre objeto y centroides, a fin de reubicarlos en el cluster cuyo centroide resulte más cercano.
Con que exista una alteración en la asignación de un elemento a un conglomerado, el centroide del mismo puede verse afectada, con lo que es necesario recalcularla.

4) Repetir el paso 3) hasta que tras un ciclo no exista ninguna modificación en cuanto a la ubicación de los elementos en los cluster.

Cabe destacar que el algoritmo de las k-modas ofrece solución óptima local independientemente de los centroides iniciales considerados y que la convergencia del algoritmo es casi segura (Huang, 1998, pg. 290).

Con respecto a la selección de los centroides iniciales para comenzar a ejecutar el algoritmo, en Huang (1998) se plantean algunas alternativas:

1.- Tomar los k primeros objetos como centroides iniciales (siempre y cuando no haya entre ellos dos elementos iguales).

2.- Proceder con los siguientes pasos:

2.1 Calcular las frecuencias de todos los atributos para cada una de las categorías y escribirlas en orden descendiente (de forma que $f(b_{i,j}) \geq f(b_{i+1,j})$):

$$\begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & & b_{33} & b_{34} \\ b_{41} & & b_{43} & \\ & & b_{53} & \end{bmatrix}$$

2.2 Definir los centroides como:

- $C_1 = [b_{11}\ ,\ b_{22}\ ,\ b_{33}\ ,\ b_{14}]$
- $C_2 = [b_{21}\ ,\ b_{12}\ ,\ b_{43}\ ,\ b_{24}]$
- $C_3 = [b_{31}\ ,\ b_{22}\ ,\ b_{13}\ ,\ b_{34}]$

La idea es tomar para C_1 los elementos de la diagonal de la matriz construida. Cuando para una variable en concreto no exista dicho elemento, se tomará el siguiente en orden circular de las categorías de dicha variable.
Para consecutivos C_{i+1} se tomará el siguiente elemento (en orden circular) de cada una de las componentes del C_i anterior.

2.3 Para evitar clusters vacíos, una vez obtenidos $C_1, \ldots, C_k$, se reemplazan por los elementos de $\mathbf{X}$ más cercanos a cada uno de estos centroides.

La diversidad de centroides que ofrece inicialmente el segundo de los métodos propuestos, provoca la obtención de mejores resultados.

Como a menudo la naturaleza de las variables de un conjunto de datos no es exclusivamente numérica ni categórica, Huang (Huang (1998)) construye un modelo para datos mixtos basado en una

ponderación de los algoritmos de las k - medias y el de las k - modas.
Básicamente se consideran los diferentes tipos de variables y se les aplica los métodos vistos según la naturaleza de las mismas.

Algoritmo k - prototipos

Cuando de datos mixtos se trata, la propuesta de Huang para definir la distancia entre dos elementos X e Y definidos a través de las variables $A_1,\ldots,A_p,A_{p+1},\ldots,A_m$ es:

$$d_2(X,Y) = \sum_{j=1}^{p}(x_j - y_j)^2 + \gamma \sum_{j=p+1}^{m} \delta(x_j, y_j)$$

donde el primer sumando se corresponde con una distancia euclídea para las variables numéricas y el segundo hace referencia a la distancia d_1 definida para variables categóricas en el algoritmo de las k - modas. El peso γ se emplea a fin ponderar la importancia de las variables categóricas tomando como referencia las variables numéricas.

A partir de la distancia d_2 definida para objetos mixtos, se puede reescribir la función costo como sigue:

$$P(W,\mathscr{C}) = \sum_{l=1}^{k}\left(\sum_{i=1}^{n} w_{i,l}\sum_{j=1}^{p}(x_{i,j}-c_{l,j})^2 + \gamma\sum_{i=1}^{n} w_{i,l}\sum_{j=p+1}^{m}\delta(x_{i,j},c_{l,j})\right).$$

Con el cambio de notación

$$P_l^r = \sum_{i=1}^{n} w_{i,l}\sum_{j=1}^{p}(x_{i,j}-c_{l,j})^2$$
$$P_l^c = \gamma\sum_{i=1}^{n} w_{i,l}\sum_{j=p+1}^{m}\delta(x_{i,j},c_{l,j})$$

se puede reescribir el problema como $P(W,\mathscr{C}) = \sum_{l=1}^{k}(P_l^r + P_l^c)$.
Debido al carácter no negativo de cada uno de los sumandos del problema, minimizar $P(W,\mathscr{C})$ es equivalente a minimizar P_l^r y P_l^c con $1 \leq l \leq k$. Para llevar a cabo los cálculos basta con proceder de la misma manera que el algoritmo presentado en el de las k - medias. Por un lado, dado un $\hat{\mathscr{C}}$, lo único que varía es la definición de distancia establecida; mientras que por otro, fijado $\hat{W}$ se calcula $\mathscr{C}$ minimizando P_l^r y P_l^c para $1 \leq l \leq k$. Con este fin, la minimización de los elementos se realizará en base a su carácter numérico o categórico por las técnicas explicadas en los algoritmos k - medias y k - modas respectivamente.

Las propuestas de Huang suponen una aportación de gran valor para extender el paradigma de las k - medias al contexto de datos categóricos. Sin embargo, al igual que en el caso del k - medias en

el contexto de datos continuos, no toman en consideración el carácter dinámico de los datos y, por consiguiente; no deben conducir a buenos resultados cuando el principio de similaridad entre objetos esté gobernado por esta característica. Los estudios de simulación mostrados en el Capítulo 2 de esta memoria corroboran este hecho.

1.2. Considerando el carácter dinámico de los datos

Uno de los motivos principales por los que falla la propuesta de Huang es la falta de consideración de las características que pueda presentar el conjunto de datos con el que se trabaja. Así, reducir la información aportada por los diversos valores que puede tomar una variable a uno único (la moda) y utilizar como distancia la SMD dan lugar a que el algoritmo de las k-modas no ofrezca resultados competitivos. Motivados por estas características, surgen posteriormente nuevas ideas al respecto. Dependiendo de las hipótesis de partida y la metodología con la que trabaja el algoritmo, éste se puede dividir en dos grandes grupos: algoritmos basados en un modelo (*model-based algorithms*) o algoritmos basados en disimilaridad (*dissimilarity-based algorithms*).
Mientras los primeros parten de la idea de que los datos fueron generados por un modelo, y tratan de recuperarlo utilizando cadenas de Markov; los segundos basan su procedimiento de clasificación o asignación cluster a partir de una métrica (tal y como los algoritmos k-medias y k-modas).

En esta sección se presentan las propuestas formuladas en los artículos más significativos referentes a los dos grandes tipos de algoritmos mencionados previamente.

1.2.1. Algoritmos basados en un modelo

Los algoritmos basados en modelos descansan en la hipótesis de partida de que la realización de los registros se ciñe a un modelo probabilístico específico. La idea básica es asumir modelos concretos (frecuentemente cadenas de Markov o mixtura de las mismas) y estimar sus parámetros. Con esta idea se construyen las propuestas de Cadez et al. (2003) y Pamminger and Frühwirth-Schnatter (2010), suponiendo el primero un modelo basado en cadenas de Markov para definir el comportamiento de los usuarios de la web `www.msnbc.com` a través de las distintas categorías de información que ofrece la página (actualidad, deportes, tiempo, viajes, página principal, ...). Para ello, gracias a las *cookies* de cada usuario y por medio de una transformación de los datos obtenidos a través de ellas, se ha simplificado la información pasando a obtener una secuencia de entradas a categorías por cada uno de los visitantes de la web.

Del modelo planteado por Cadez et al., cabe destacar que:

- capta el dinamismo de la información debido a que las variables aleatorias hacen referencia a categorías visitadas dentro de un mismo itinerario de navegación.
- gracias a la suposición de que un modelo de Markov distinto es el que rige cada uno de los conglomerados, el planteamiento consigue captar la heterogeneidad existente en los distintos clusters debida a la diversidad de la duración de la estancia en cada categoría, los contenidos visitados, . . .
- los datos tratados en los ejemplos del artículo son modelizados como si se hubiesen generado de la siguiente manera.

  1) Un usuario llega a la red y se le asigna a un cluster con cierta probabilidad.
  2) El comportamiento del usuario se genera a partir de un modelo de Markov con parámetros específicos de ese grupo. Así pues, debe de estar concretado el comportamiento de los individuos para cada uno de los conglomerados.

- para determinar la proporción de usuarios existentes en cada cluster y especificar los parámetros de cada modelo de Markov, se utiliza el algoritmo EM , descrito más adelante.

De gran relevancia es también, el hecho de que el algoritmo de Cadez et al. (2003) sea más eficiente computacionalmente que otros propuestos hasta la fecha y que ni siquiera logran captar las características mencionadas. Así, mientras que un algoritmo jerárquico resulta ser del orden $O(n^2)$ (siendo n el número de datos totales), el presentado en el artículo es del orden de $O(knt)$ siendo k el número de conglomerados y t el número medio de componentes de cada una de las secuencias que definen al usuario. Como a lo largo del presente trabajo se considera que un usuario está definido por una secuencia de m valores, el parámetro t anterior se correspondería con el valor m .
La mejora es importante puesto que el número de conglomerados k siempre va a ser mucho menor que la cifra de datos totales y el número medio de componentes t no acostumbrará a ser demasiado extenso. Así, cuando la dimensionalidad de los datos sea grande, los algoritmos presentados mejorarán en tiempo de ejecución a los jerárquicos.

A continuación, se presenta la estructura que sigue el modelo propuesto por los autores y formas de estimación de los parámetros requeridos para el modelo en base a esa configuración.

El planteamiento del algoritmo asume independencia a la hora de generar los valores que definen a cada usuario. Para la formación de uno en particular se considera la llegada de un individuo a la web y su asignación (definida a partir de una ley de probabilidad) a un grupo en concreto. Una

vez determinado el cluster de pertenencia, su comportamiento viene definido a través de un modelo estadístico (cadena de Markov) propio del conglomerado al que pertenece.

Con estas suposiciones acerca de la estructura, los objetivos del algoritmo son calcular el número de conglomerados, hallar la función de probabilidad que asigna un usuario a un cluster en particular y determinar los parámetros de cada conglomerado (funciones de probabilidad, funciones de transición, ...).

Suponiendo que esta composición (denominada estructura de modelos mixtos) consta de k conglomerados, se puede considerar que la probabilidad de que conocidos los parámetros θ del modelo, se obtenga el elemento X viene determinada por:

$$p(X \mid \theta) = \sum_{l=1}^{k} p(c_l \mid \theta) p_l(X \mid c_l, \theta),$$

donde:

- $X = [x_1, \ldots, x_m]$ representa un individuo genérico resultado de la concatenación de valores para cada una de las variables consideradas en el análisis. Así, X determina un elemento del conjunto de datos, con valores concretos x_i para cada una de las categorías A_i siendo $i \in \{1, \ldots, m\}$.
- θ hace referencia al conjunto de parámetros que describen el modelo considerado.
- los $c_l : l \in \{1, \ldots, k\}$ se consideran indicadores de cada uno de los k conglomerados supuestos para el modelo.
- $p(c_l \mid \theta)$ es la probabilidad marginal de que un elemento pertenezca al l - ésimo cluster. Como consecuencia inmediata se tiene que $\sum_{l=1}^{k} p(c_l \mid \theta) = 1$.
- $p_l(X \mid c_l, \theta)$ es el modelo estadístico que describe la distribución para las variables dentro del l - ésimo cluster; es decir la probabilidad de obtener el individuo X sabiendo que pertenece al conglomerado especificado.

Esta sucesión de m valores categóricos definen la secuencia de secciones que ha visitado el usuario de forma cronológica. Como consecuencia, no se considerarán más allá de m instantes en los que se determine la ubicación del usuario.
Asumiendo que cada cluster se rige por un proceso de Markov de primer orden, la probabilidad de obtener un elemento concreto X sabiendo que éste pertenece al l - ésimo cluster viene dada por la formulación siguiente:

$$p_l(X \mid c_l, \theta) = p(x_1 \mid \theta_l^I) \prod_{i=2}^{m} p(x_i \mid x_{i-1}, \theta_l^T),$$

siendo

- θ_l^I los parámetros de la función de probabilidad sobre las categorías de la página inicial para un usuario del l - ésimo cluster.
- θ_l^T la matriz de dimensión $m\,x\,m$ definiendo las probabilidades de transición de una categoría a otra para un usuario perteneciente al l - ésimo conglomerado.

Esta modelización mediante cadenas de Markov para cada cluster permite que el modelo recoja información a cerca de: el orden de interés o navegación del usuario (hasta cierto punto), el valor inicial o categoría de entrada en la red, la dependencia entre accesos consecutivos del usuario, la última categoría visitada, ….

En lugar de suponer una cadena de Markov de primer orden para cada conglomerado, existen otras opciones. Considerarlas de un orden mayor permite captar la dependencia entre etapas más distantes en el tiempo, si bien el modelo resulta ser más complejo y costoso en cuanto a la estimación de los parámetros.
Otra posibilidad es simplificar el modelo, tomando cadenas de Markov de orden cero. Su modelización vendría dada por:

$$p_l(X \mid c_l, \theta) = \prod_{i=1}^{m} p(x_i \mid \theta_l^M),$$

donde θ_l^M se refiere a los parámetros de la distribución marginal sobre la categoría requerida en el l - ésimo cluster. Este método no da tanta trascendencia a la primera entrada del usuario como lo hacía un modelo de Markov de primer orden. Es más, se le otorga la misma importancia a cada uno de los accesos a secciones de la web, con lo que resulta un modelo útil cuando se recogen las rutas de navegación sin tener en cuenta el orden de acceso de las mismas.

Si se trabaja con un modelo que presupone que cada cluster está definido a través de una cadena de Markov y a mayores se tiene conocimiento al respecto de éstas gracias a que el proceso ha sido sometido a aprendizaje, esa información puede ser utilizada para la asignación de un usuario a un grupo en particular. Para ello, dado un individuo X; a través de la regla de Bayes se puede determinar la probabilidad de que éste pertenezca a un cluster particular. Para ello basta calcular $\forall\, l \in \{1,\ldots,k\}$ la probabilidad $p(c_l \mid X, \theta)$ dada por:

$$p(c_l \mid X, \theta) = \frac{p(c_l \mid \theta) p_l(X \mid c_l, \theta)}{\sum_{l'=1}^{k} p(c_{l'} \mid \theta) p_{l'}(X \mid c_{l'}, \theta)}.$$

En base a estas probabilidades denominadas *membership probabilities*, se pueden hacer dos tipos de asignación a clusters:

- *Hard assignment*: en ella se le asigna al usuario el conglomerado respecto del cuál es más probable que haya sido originado.
- *Soft assignment*: en ella se tienen en consideración cada una de las las k probabilidades de pertenencia a cada conglomerado. Un ejemplo de ello es el *fuzzy clustering*, donde cada elemento tiene un grado de pertenencia difuso definido por el *soft assignment*.

Referente a la estructura del modelo y para poder realizar una asignación de los usuarios a un cluster; en cualquiera de los casos es necesario conocer previamente los parámetros θ que lo definen. Al respecto, existen varios métodos de estimación. Los mencionados a continuación utilizan una muestra de entrenamiento $d_{train} = \{x^1, \ldots, x^N\}$ para realizar el aprendizaje de los parámetros de un modelo mixto con k clusters.

Uno de los criterios posibles para la estimación de los parámetros consiste en tomar los valores de θ que maximicen la verosimilitud (ML, *maximum likelihood* de la muestra d_{train}. Así se obtendría:

$$\theta^{ML} = arg\max_{\theta} p(d_{train} \mid \theta) = arg\max_{\theta} \prod_{i=1}^{N} p(x^i \mid \theta).$$

Alternativamente, para codificar el conocimiento previo que pueda existir sobre el dominio o para suavizar las estimaciones obtenidas mediante máxima verosimilitud, se puede introducir una distribución a priori sobre los parámetros ($p(\theta)$). Éste es el criterio sobre el que se apoya la estadística bayesiana, que busca maximizar la probabilidad a posteriori (MAP, *maximum a posteriori*) del valor del parámetro. Ante esta situación, un criterio para determinar los valores del modelo consiste en identificar aquellos que maximicen esta probabilidad para el parámetro θ dada la muestra de entrenamiento. Utilizando el teorema de Bayes resulta:

$$\theta^{MAP} = arg\max_{\theta} p(\theta \mid d_{train}) = arg\max_{\theta} \frac{p(d_{train} \mid \theta)p(\theta)}{p(d_{train})}.$$

En cualquier caso, la estimación de los parámetros se realiza mediante la maximización de las correspondientes fórmulas. Dicha labor no es inmediata y es por ello que es necesario su cálculo mediante algoritmos como el EM.

Implementación del algoritmo EM

Sea $d_{train} = \{x^1, \ldots, x^N\}$ la muestra de entrenamiento consistente en N individuos definidos como $x^i = \{x^i_1, \ldots, x^i_m\}$. Cada uno de los estados x^i_j puede tomar valores de una variable discreta $\{1, \ldots, M\}$ que se correspondería con las categorías de la página web. El parámetro θ viene representado por la terna $\theta = \{\pi, \theta^I, \theta^T\}$ donde:

- π es un vector formado por k pesos que determinan la probabilidad de pertenecer a cada cluster

$$\pi = \{\pi_1, \ldots, \pi_k\} \;\; : \;\; \pi_l = p(c_l \mid \theta) \;\; \Rightarrow \;\; \sum_{l=1}^{k} \pi_l = 1.$$

- θ^I se corresponde con el conjunto de vectores que determinan para cada conglomerado la probabilidad de que la primera visita sea a una sección determinada. Así, θ^I es un conjunto de vectores

$$\theta^I = \{\theta^I_1, \ldots, \theta^I_k\}$$

  siendo cada uno de ellos definido como:

$$\theta^I_l = \{\theta^I_{l,1}, \ldots, \theta^I_{l,M}\} \;\; : \;\; \theta^I_{l,j} = p(x_1 = j \mid c_l, \theta) \;\; \Rightarrow \sum_{j=1}^{M} \theta^I_{l,j} = 1 \;\; \forall\, l \in \{1, \ldots, k\}.$$

- θ^T es un conjunto de k matrices de transición; una para cada cluster.

$$\theta^T = \{\theta^T_1, \ldots, \theta^T_k\}$$

  Cada matriz θ^T_l tiene dimensión M x M y sus elementos se corresponden con la probabilidad de pasar de un estado j a otro estado h , dentro del k - ésimo conglomerado.

$$\theta^T_l = \left[\theta^T_{l,j,h}\right] \;\; : \;\; \theta^T_{l,j,h} = p(x_t = h \mid x_{t-1} = j,\, c_l, \theta) \;\; \Rightarrow \;\; \sum_{h=1}^{M} \theta^T_{l,j,h} = 1 \;\; \forall\, l, j.$$

Se define la distribución de probabilidad clase - condicionada (*class-conditional probability distribution*) como la probabilidad de que la secuencia x^i haya sido generada por el l - ésimo cluster dados los parámetros θ:

$$P_{i,l}(\theta) = p(c_l \mid x^i, \theta) = \frac{\pi_l p(x^i \mid c_l, \theta)}{\sum_{l'=1}^{k} \pi_{l'} p(x^i \mid c_{l'}, \theta)}.$$

Estos valores pueden ser recogidos en una matriz P tal que

$$P(\theta) = \left[P_{i,l}(\theta)\right] \quad 1 \leq i \leq N, \quad 1 \leq l \leq k.$$

El algoritmo Expectation-Maximization (EM) es un método para el cálculo iterativo de estimaciones máximo-verosímiles (MV) en una amplia variedad de problemas. En cada iteración del algoritmo, hay dos pasos: un Expectation-step (o E-step) y un Maximization-step (o M-step). A cada paso se actualizan los valores de los parámetros ($\pi,\ \theta^I,\ \theta^T$) a partir de la maximización de la función correspondiente al tipo de estimación de parámetros efectuada (ML o MAP).

En Pamminger and Frühwirth-Schnatter (2010) se mencionan dos alternativas para el análisis cluster de series de tiempo con respuesta categórica. La primera de ellas denominada *Markov chain clustering* se basa en las mismas suposiciones que Cadez et al. (2003); un modelo de Markov rige cada conglomerado y todos los elementos del mismo se adecúan a sus funciones de transición. La segunda opción propuesta consiste en una mezcla finita de modelos de efectos aleatorios a fin de captar la heterogeneidad que se supone que existe entre transiciones dentro de un mismo cluster. Se trata simplemente de una pequeña variación en el patrón que rige cada conglomerado a fin de dar mayor libertad a los individuos del mismo, si bien de una forma global todos ellos presentan comportamientos similares. Este algoritmo denominado *Dirichlet multinomial clustering* por los autores, debe su nombre a que es una distribución de Dirichlet la que rige las pequeñas desviaciones de los usuarios de un mismo cluster respecto de un comportamiento común.
En Pamminger (2012), Pamminger implementa un paquete para el lenguaje de programación R que incluye las dos propuestas de algoritmos mencionadas en Pamminger and Frühwirth-Schnatter (2010). El primero de los modelos (*Markov chain clustering*) será ejecutado posteriormente en las simulaciones propuestas en el Capítulo 2 de este documento.

1.2.2. Algoritmos basados en disimilaridad

Se caracterizan por la utilización de una métrica entre los individuos en base a la cuál se desarrolla el proceso de formación de conglomerados. La definición de dicha medida es crucial en el proceso de clustering. Obviamente, su construcción atenderá a reflejar el concepto de disimilaridad que se pretende que gobierne el desarrollo de la formación de grupos. De esta forma, aunque diferentes criterios de disimilaridad conducirán a distintas soluciones, el adecuado será aquél que mejor refleje el criterio de distancia deseado. Conviene destacar que, una vez introducida la disimilaridad, algoritmos cluster estándar pueden ser aplicados para determinar los conglomerados, sin necesidad de concretar

parámetros adicionales como en el caso de los procedmientos basados en modelos. A continuación, se presentan dos procedimientos basados en disimilaridad propuestos recientemente en la literatura.

1.2.2.1. Algoritmo de Ahmad y Dey

El artículo Ahmad and Dey (2007) destaca por la propuesta de una distancia entre elementos que tiene en cuenta la distribución completa de las variables en vez de simplemente el valor más frecuente de las mismas (como hacía el algoritmo de las k - modas). Trata de aprovechar toda la información sobre cada variable básicamente modificando la función costo (llámese también función pérdida) y la distancia entre elementos.

La función pérdida propuesta en el artículo viene dada por:

$$P(W,\mathscr{C}) = \sum_{i=1}^{n} \vartheta(x_i,\mathscr{C})$$

donde

$$\vartheta(x_i,\mathscr{C}) = \sum_{l=1}^{k} w_{i,l} \left(\sum_{j=1}^{p} (\alpha_j(x_{i,j} - c_{l,j}))^2 + \sum_{j=p+1}^{m} \Omega(x_{i,j}, c_{l,j})^2 \right)$$

está formado por la suma de la distancia con respecto del cluster más cercano, separando la parte numérica de la categórica. Mientras la primera hace referencia a una ponderación de la distancia euclídea (mediante los α_j), la segunda resulta ser una nueva definición de distancia. Su definición y la forma de obtener las ponderaciones α_j serán introducidas posteriormente.

Algunas consideraciones y diferencias con respecto del planteamiento propuesto por Huang (1998) son las siguientes:

- Ahmad y Dey proponen una estandarización de las variables numéricas previa al análisis cluster, a fin de tratarlas con la misma escala.
- a diferencia de Huang, que pondera con la misma importancia a cada una de las variables; Ahmad y Dey otorgan mayor importancia a aquellas que facilitan la separación de elementos en términos de distancias. Los valores α_j que ponderan a las variables son calculados a partir del conjunto de datos, con lo que no existe ningún problema al respecto de su obtención.
- la definición de distancia se ve alterada para datos categóricos. La distribución global de la variable es tomada en cuenta y las coocurrencias entre valores para las mismas desempeñan un papel importante.

- para cada atributo, Huang define el centroide como la concatenación de valores específicos (medias o medianas) en función del carácter de las variables. Ahora, cada componente del centroide será:

  - la media de los valores que toma la variable si se trata de una numérica.
  - la distribución de la variable si ésta es categórica.

Con respecto a los cambios propuestos por los autores, la distancia considerada trata de captar las coocurrencias más repetidas entre valores distintos de una misma variable. A lo largo de la definición de la distancia se utilizará en la notación:

- A_i como variable categórica que toma varios valores entre los que se encuentran x e y.
- A_j como otra variable categórica de interés.
- w refiriéndose a un subconjunto de los posibles valores que toma la variable A_j. Así, $\bar{w}$ denotará el conjunto complementario de w.
- $P_i(w \mid x)$ determina la probabilidad de que un elemento con x como respuesta para la variable A_i presente un valor del conjunto w para la variable A_j. De igual manera $P_i(\bar{w} \mid y)$ representa la probabilidad de que un elemento con y como valor para la variable A_i presente un valor para A_j que pertenezca al conjunto $\bar{w}$.

En base a esta notación, se define la distancia entre un par de valores x e y con respecto al subconjunto w de la variable A_j:

$$\delta_w^i(x,y) = P_i(w \mid x) + P_i(\bar{w} \mid y).$$

Y así, la distancia general entre x e y (respuestas de la variable A_i) con respecto de la variable A_j como:

$$\delta^{ij}(x,y) = P_i(\omega \mid x) + P_i(\bar{\omega} \mid y) - 1,$$

donde ω es el subconjunto de valores de A_j que maximiza la suma $P_i(\omega \mid x) + P_i(\bar{\omega} \mid y)$.

La resta de una unidad es debida a la intención de que $\delta^{ij}(x,y)$ sea un valor entre 0 y 1. La suma $P_i(\omega \mid x) + P_i(\bar{\omega} \mid y)$ siempre será mayor o igual que uno. Este resultado lo garantiza la posibilidad de tomar el conjunto vacío o el total como valor de ω. Con esta definición de distancia se pretende capturar la máxima aportación que ofrecen los valores x e y de la variable A_i en el proceso de clustering, suponiendo la variable A_j como único atributo existente a parte de A_i.

Sólo se tiene en cuenta la relación de los valores x e y con respecto de una única variable A_j. No obstante en la definición de distancia entre dos elementos, intervienen todas y cada una de las variables que definen al usuario.
Dado un conjunto de datos en el que cada individuo viene definido a partir de m variables aleatorias (bien de carácter categórico o mixto), la distancia entre dos valores distintos x e y de cierta variable categórica A_i viene dada como el promedio de cada una de las distancias de los valores con respecto del resto de variables:

$$\delta(x,y) = \frac{1}{m-1} \sum_{j=1}^{m} \sum_{j \neq i} \delta^{ij}(x,y). \tag{1.1}$$

La definición de distancia mostrada es de uso exclusivo para valores de variables categóricas. No obstante, en esta definición intervienen también las variables numéricas. Para ello, es necesario discretizarlas a fin de poder computar las distancias.

Cabe destacar que pese a que dos variables A_i y A_j tomen los mismos valores, la distancia entre dos elementos (x e y) de las mismas puede ser distinta. Ésto es debido a que en la distancia entre los elementos intervienen las distribuciones que presentan el resto de variables, situación que no tiene por qué ser idéntica en todos los casos. Con respecto al número de cuentas, suponiendo que cada variable A_i tiene M posibles valores respuesta, y que existen m variables aleatorias, sería necesario calcular $m \frac{M(M-1)}{2}$ distancias entre elementos debido a propiedades de la distancia definida, como:

- $\delta(x,x) = 0 \ \ \forall\, x.$
- $\delta(x,y) = \delta(y,x) \ \ \forall\, x,y.$
- $0 \leq \delta(x,y) \leq 1.$

Con esta definición de distancia entre dos valores categóricos de una misma variable aleatoria, cuanto mayor sea entre x e y; mayor es la probabilidad de que datos con dichos valores (para la variable específica en la que se halla calculado la distancia) pertenezcan a conglomerados distintos. Es por ello, que en la ejecución del algoritmo propuesto por Ahmad y Dey resulta muy importante el cálculo de estas distancias.

A parte de esta medida de disimilaridad, otra de las novedades que presentaba la función pérdida propuesta por Ahmad y Dey en relación con la diseñada por Huang, era la introducción de unos valores de significación para las variables numéricas. Estos pesos pretenden determinar la trascendencia de cada variable en el conjunto de datos. Esa importancia viene cuantificada de forma que si existen

valores de la variable que disten mucho entre sí (resultando útiles de cara al análisis cluster), el factor de la variable numérica sea mayor que otras variables aleatorias que no sean de tanta utilidad al proceso.

Con la definición de distancia para valores de variables categóricas, éstas ya llevan de una forma intrínseca el valor de significación del atributo. Las variables numéricas sin embargo, no presentan un coeficiente que determine su trascendencia, por lo que se propone calcularlo tal y como se indica a continuación. Es necesaria una discretización de las variables numéricas en intervalos para poder determinar su significación. Esta partición únicamente será utilizada para calcular el α_j correspondiente a cada variable numérica A_j.

El cálculo de los α_j se realiza siguiendo estos pasos:

1) Estandarizar la variable numérica A_j.

2) Dividir la variable en S intervalos $(u[1],\ldots,u[S])$. Si bien no se especifica qué valor debe de ser considerado como S, puede resultar de interés investigar sobre su trascendencia en el algoritmo.

3) Hallar $\delta(u[v_1],u[v_2])$ para cada par (v_1,v_2) con $v_1,v_2 \in \{1,\ldots,S\}$ de la misma forma que se realizaba para elementos de variables categóricas.

4) Finalmente, se define la significación de la variable numérica A_j como el promedio de las distancias entre los intervalos $u[1],\ldots,u[S]$.

$$\alpha_j = \frac{1}{\binom{S}{2}} \sum_{v_1=1}^{S} \sum_{v_2>v_1}^{S} \delta(u[v_1],u[v_2]).$$

Distancia entre dos individuos

A partir de la distancia definida para dos valores respuesta de una misma variable y la significación de las variables numéricas, se puede definir la distancia entre dos individuos. Suponiendo dos de forma genérica $X_1 = [x_{11},\ldots,x_{1m}]$ y $X_2 = [x_{21},\ldots,x_{2m}]$, Ahmad y Dey proponen como distancia entre ambos la definida por:

$$d_3(X_1,X_2) = \sum_{j=1}^{p} (\alpha_j(x_{1j}-x_{2j}))^2 + \sum_{j=p+1}^{m} (\delta(x_{1j},x_{2j}))^2,$$

donde las p primeras variables son numéricas y las restantes son categóricas.

A partir de la definición de distancia entre dos individuos se construye una matriz simétrica de dimensión $n \times n$ que recoja las distancias entre todos los elementos de la base de datos de estudio.

Centro de un cluster

Tal y como se habrá observado en las secciones previas, el cómputo de un centro (media, centroide, moda,...) para cada cluster es un paso común a todos los algoritmos partitivos de clustering. El planteamiento de Ahmad y Dey varía con respecto a métodos como el de las k - modas, pretendiendo recoger la mayor información posible. Así, el cálculo de los centroides se realiza de la siguiente manera.

- Si todas las variables consideradas son numéricas, el centroide de un conglomerado tendrá por j-ésima componente la media de la variable A_j sobre las observaciones de dicho conjunto.
- En el momento en el que los datos presentan alguna variable categórica:
  - la componente de una variable numérica es la media de todos los datos del conglomerado para dicha variable.
  - la componente de una variable categórica está formada por la distribución de frecuencias de todos los valores posibles que pueda tomar. Las frecuencias vienen concretadas por las proporciones de observaciones para cada valor de la variable dentro del cluster.

Para un cluster particular de N_l elementos en el que los datos vienen representados de la forma $X_i = \left[x_{i1}, \ldots, x_{ip},\ x_{i(p+1)}, \ldots, x_{im}\right]$ (donde los p primeros atributos son numéricos y los $m-p$ restantes categóricos), el centroide vendría determinado por:

$$C_l = \left[\frac{\sum_{i=1}^{N_l} x_{i,1,l}}{N_l}, \ldots, \frac{\sum_{i=1}^{N_l} x_{i,p,l}}{N_l},\ \frac{(N_{l,(p+1),1}, \ldots, N_{l,(p+1),m_{(p+1)}})}{N_l}, \ldots, \frac{(N_{l,m,1}, \ldots,\ N_{l,m,m_m})}{N_l} \right],$$

siendo $N_{l,i,j}$ el número de individuos en el l-ésimo conglomerado que presentan la j-ésima respuesta para la i-ésima variable (que tiene m_i posibles respuestas). Valga también a fin de abreviar notación la siguiente expresión:

$$C_l = \left[c_{l,1}, \ldots, c_{l,p}, \left[c_{l,(p+1),1}, \ldots, c_{l,(p+1),m_{(p+1)}} \right], \ldots, \left[c_{l,m,1}, \ldots, c_{l,m,m_m} \right] \right]$$

donde cada término $c_{l,i}$ $i \in \{1, \ldots, p\}$ hace referencia a la media para la i-ésima variable numérica del l-ésimo cluster y cada componente $c_{l,i,j}$ $i \in \{p, \ldots, m\}$ $j \in \{1, \ldots, m_i\}$ supone la proporción de elementos existentes en el l-ésimo conglomerado que presentan la j-ésima respuesta para la i-ésima variable aleatoria.

Distancia entre individuo y centroide

Otro de los pasos importantes en el algoritmo es el cálculo de las distancias entre individuos y cada uno de los centroides existentes. Así, el cluster al que pertenezca el centroide más cercano a un individuo; será el conglomerado al que será asignado. Con la nueva definición de centroide, la distancia debe de ser redefinida ya que en las variables categóricas no se tiene un único elemento, sino las frecuencias relativas.

Considérese un individuo $X_i = \left[x_{i,1}, \cdots, x_{i,p}, x_{i,(p+1)}, \cdots, x_{i,m}\right]$ y un centroide genérico

$$C_l = \left[c_{l,1}, \ldots, c_{l,p}, \left[c_{l,(p+1),1}, \ldots, c_{l,(p+1),m_{(p+1)}}\right], \ldots, \left[c_{l,m,1}, \ldots, c_{l,m,m_m}\right]\right].$$

Denotando por N_l el número de individuos en el l-ésimo cluster, $x_{i,j}$ la j-ésima componente del individuo X_i e $y_{l,j,h}$ la h-ésima respuesta posible para la j-ésima vairable en lo que al l-ésimo conglomerado concierne, se define la distancia entre un individuo X_i y el centroide C_l como:

$$\sum_{j=1}^{p} (\alpha_j(x_{i,j} - c_{l,j}))^2 \; + \; \sum_{j=p+1}^{m} \Omega(x_{i,j}, c_{l,j})^2,$$

con $\Omega(x_{i,j}, c_{l,j}) = \sum_{h=1}^{m_j} \frac{N_{l,j,h}}{N_l} \delta(x_{i,j}, y_{l,j,h})$.

Para cuantificar la distancia, la parte correspondiente a las variables numéricas se determina mediante la distancia euclídea. La correspondiente a las variables categóricas se expresa como un sumatorio de las distancias entre el valor que toma el inviduo con respecto de todos los valores posibles, ponderados por la frecuencia relativa que presentan las opciones de respuesta dentro del cluster.

Habiendo establecido las distancias entre dos individuos y entre un individuo y un centroide, el algoritmo cluster consta de las siguientes etapas:

Algoritmo de clustering (Ahmad y Dey)

1) Para cada variable categórica hallar las distancias entre cada una de las respuestas posibles.

2) Para cada variable numérica hallar el valor de significación de la misma.

3) Ubicar de forma aleatoria a cada individuo en uno de los k clusters.

4) Repetir los pasos siguientes hasta que ningún elemento cambie de grupo (en dos etapas consecutivas) o hasta que se efectúe un número concreto de iteraciones:

   4.1) Hallar los centroides de cada conglomerado.

   4.2) Asignar de nuevo cada elemento al cluster cuyo centroide se encuentre más próximo.

Conviene destacar con respecto del algoritmo de Ahmad y Dey que la calidad del resultado obtenido depende del número de clusters impuesto. La convergencia del algoritmo no es exclusiva a un escenario en particular, si bien existen algunas sugerencias de inicialización del algoritmo (Bradley and Fayyad (1998)). Aún así, Pena et al. (1999) demuestran que la distribución aleatoria de los individuos a conglomerados, da lugar a un algoritmo más efectivo. Sobre los tiempos de ejecución de las distintas fases del algoritmo se tiene:

- Distancia entre dos valores específicos de una variable: $O(m^2n + m^2M^3)$.
- Ejecución de un paso del algoritmo (cáclulo de centroides más asignación de los elementos a un conglomerado): $O(nkp + nk(m-p)M)$.
- Ejecución del algoritmo para e etapas: $O(m^2n + m^2M^3 + en(kp + k(m-p)M)$.

donde n es el número de individuos, m es el número total de variables consideradas (p de ellas numéricas y las $(m-p)$ restantes categóricas), M es el número de respuetas de cada variable (considerando todas con el mismo número) y k es el número de clusters impuestos por el usuario.

1.2.2.2. Algoritmo de García-Magariños y Vilar

La distancia de Ahmad y Dey tiene en cuenta la coocurrencia de respuestas categóricas al medir similaridad entre individuos, lo que en buena lógica debería de ayudar a detectar parecidos entre conductas dinámicas de los mismos. Sin embargo, no se tiene en cuenta el orden cronológico en que esas coocurrencias acontecen. En García-Magariños and Vilar (2014) se propone una nueva métrica de disimilaridad fundamentada en medir simultáneamente discrepancia entre valores en bruto de las respuestas y sus evoluciones temporales. Específicamente, la disimilaridad introducida tiene la forma general $\phi(Corr(x,y))d(x,y)$, donde $d(x,y)$ es una distancia estática, midiendo disimilaridad entre registros en bruto, $Corr(x,y)$ es una medida de correlación captando el grado de similaridad entre conductas a lo largo del tiempo y ϕ es una función que regula el peso de esta correlación en la disimilaridad final mediante la modificación de un parámetro específico definido más adelante.

Conviene destacar que en García-Magariños and Vilar (2014) los usuarios genéricos vienen determinados exclusivamente por variables con respuesta categórica. Así, se considerarán cada una de las m variables como tal.

Con respecto a la distancia entre los usuarios $X_1 = [x_{11}, \ldots, x_{1m}]$ y $X_2 = [x_{21}, \ldots, x_{2m}]$ los autores proponen utilizar la distancia de Ahmad y Dey de modo que:

$$d_3(X_1,X_2) = \frac{1}{m}\sum_{j=1}^{m}\delta(x_{1j},x_{2j}).$$

donde la función $\delta(x_{1j},x_{2j})$ se define como en la Ecuación 1.1.

La correlación que se introduce en esta proposición, pretende ser insertada mediante un coeficiente al estilo del de correlación de Pearson. Para datos numéricos, su valor está definido, si bien debe de ser reconstruido para datos categóricos. En primer lugar habría que extender el concepto de diferencia de primer orden, $x_{(j+1)} - x_j$. Suponiendo x_j y $x_{(j+1)}$ valores fijos para las entradas j y *(j+1)*-ésima respectivamente, la idea es que la divergencia entre ambos tome valores positivos cuando la transición $x_j \to x_{(j+1)}$ sea más frecuente de lo esperado en base a las frecuencias marginales para las entradas j y $j+1$. Por consiguiente, si el valor es menor que el esperado, la cifra deberá de ser negativa.

Sean $e_j = \frac{\#\{x_j\,:\,j\}}{n}$ y $e_{j+1} = \frac{\#\{x_{j+1}\,:\,j+1\}}{n}$ las frecuencias marginales de x_j y x_{j+1} en las entradas *j*-ésima y *(j+1)*-ésima respectivamente. Siguiendo la idea de la discrepancia chi - cuadrado y la presentada previamente en referencia al signo, se define la divergencia entre los valores categóricos x_j y x_{j+1} en la transición $j \to j+1$ como:

$$Div(j,x_j,x_{j+1}) = \frac{Ob(j,x_j,x_{j+1}) - Ex(j,x_j,x_{j+1})}{Ex(j,x_j,x_{j+1})},$$

donde $Ob(j,x_j,x_{j+1}) = \#\{x_j \to x_{j+1}\ :\ j \to j+1\}$ y $Ex(j,x_j,x_{j+1}) = n\,e_j\,e_{j+1}$. De esta forma, la divergencia entre dos valores x_j y x_{j+1} tomará valores grandes y positivos cuanto más frecuente (con respecto a lo esperado) sea la transición de x_j a x_{j+1} y será negativa cuando sea menos frecuente de lo esperado.

Supóngase X_1 y X_2 como dos elementos genéricos, el siguiente paso es adaptar el concepto de covarianza. En el contexto numérico, el producto $(x_{1(j+1)} - x_{1j})\,(x_{2(j+1)} - x_{2j})$ mide la evolución de los vectores X_1 y X_2 de forma que toma valores positivos cuando ésta resulta simultánea (crecen o decrecen a la vez); presenta valores negativos cuando la evolución es de carácter opuesto (el crecimiento de uno de ellos, conlleva al decrecimiento del otro) y ofrece valores cercanos a 0 cuando no existe relación lineal entre la evolución.

Estas propiedades pueden obtenerse también con datos categóricos definiendo la covarianza entre dos individuos X_1 y X_2 como sigue:

$$Cov(X_1,X_2) = \sum_{j=1}^{m-1} Cov(j,X_1,X_2),$$

siendo definida la covarianza en la *j*-ésima entrada como:

$$Cov(j,X_1,X_2) = Div(j,x_{1j},x_{1(j+1)})Div(j,x_{1j},x_{2(j+1)}) + Div(j,x_{2j},x_{1(j+1)})Div(j,x_{2j},x_{2(j+1)}).$$

El primer sumando en la expresión anterior mide la verosimilitud de que el individuo X_1 siga el camino de X_2 en la transición de j a $j+1$. Si es positivo, esta verosimilitud es destacable en tanto que si es negativo se corresponde con que esa transición sea menos verosímil de lo esperado. Análogamente, el segundo sumando mide la versoimilitud de que X_2 siga el mismo comportamiento que X_1 en esa transición.

En el caso numérico, la correlación se acota a valores del intervalo $[-1,1]$; para lo cuál es dividida entre las desviaciones estándar. En el contexto categórico, la varianza de un elemento genérico $X = [x_1,\ldots,x_m]$, se define de la siguiente manera:

$$Var(X) = \sum_{j=1}^{m-1} 2Div^2(j,x_j,z_{j+1}),$$

donde

$$z_{j+1} = arg \max_{t \in T_{j+1}} Div(j,x_j,t)$$

para $T_{j+1} = \{t \ : \ \exists X \ , \ t = x_{j+1}\}$. A diferencia del caso numérico, en el categórico $Cov(X,X) \neq Var(X)$.

Una vez introducidos los elementos necesarios, se define la correlación entre los elementos X_1 y X_2 como el cociente entre su covarianza y el producto de sus desviaciones estándar (de forma análoga al caso numérico):

$$Corr(X_1,X_2) = \frac{Cov(X_1,X_2)}{\sqrt{Var(X_1)}\sqrt{Var(X_2)}}.$$

La definición de correlación entre dos individuos junto con la definición de distancia entre elementos propuesta por Ahmad y Dey, permite a García-Magariños y Vilar construir un índice que tenga en cuenta ambas características. Así, la proximidad temporal y la distancia entre los individuos X_1 y X_2 viene dada por:

$$d_{corr}(X_1,X_2) = \Phi_k(Corr(X_1,X_2))\ d_3(X_1,X_2), \tag{1.2}$$

siendo

$$\Phi_k(x) = \tfrac{2}{1+exp(kx)}.$$

$\Phi_k(x)$ desempeña una función reguladora en la ponderación entre correlación y distancia de forma que $d_{corr}(X_1, X_2)$ se ve reducida por Φ_k cuando la correlación sea positiva y aumenta cuando la correlación es negativa. Además, el valor de k (número natural) proporciona más importancia a la correlación cuanto más grande sea. Por experimentación, García-Magariños y Vilar determinan que $k = 5$ resulta adecuado para el análisis cluster.

Algoritmo de clustering (García-Magariños y Vilar)

El algoritmo, ejecutable mediante la consecución de los siguientes pasos, requiere inicialmente del cómputo de las distancias $d_{corr}(X_i, X_j) \ \ \forall i, j \in \{1, \ldots, n\}$. Una vez calculadas las mismas, se continúa de la siguiente forma:

Asignación inicial

La asignación inicial basada en k - medias consiste en especificar los k centroides iniciales que representan cada cluster como aquellos elementos más alejados entre sí. Si de dos conglomerados se tratase, esta opción asignaría como centroides iniciales de los clusters a los dos elementos más distantes entre sí.
Existen alternativas como una elección aleatoria, pero está comprobado empíricamente que funciona mejor la primera idea (García-Magariños and Vilar, 2014). Una vez concretados, el resto de individuos son asignados al conglomerado con respecto del cuál su elemento referencia inicial esté más cerca.

Centroide de cada cluster

Al igual que la propuesta de Ahmad y Dey, para un cluster particular de N_l elementos representados como $X_i = [x_{i1}, \ldots, x_{im}]$ (recordar que todas las variables consideradas son categóricas), siguiendo la notación previa el centroide vendría determinado por:

$$C_l = \left[\left[c_{l,1,1}, \ldots, c_{l,1,m_1} \right], \ldots, \left[c_{l,m,1}, \ldots, c_{l,m,m_m} \right] \right].$$

Con el fin de simplificar la notación en la formulación siguiente, se redefine el centroide del l-ésimo cluster como:

$$C_l = \left[c_{l,1}, \ldots, c_{l,m} \right]$$

donde

$$c_{lj} = \{(c_{lj1}, p_{lj1}), \ldots, (c_{lj(m_j)}, p_{lj(m_j)})\}$$

entendiendo por $\{c_{lj1},\ldots,c_{lj(m_j)}\}$ los m_j valores posibles que puede tomar la j-ésima variable y $\{p_{lj1},\ldots,p_{lj(m_j)}\}$ sus correspondientes probabilidades (definidas en base a los elementos que forman el conglomerado). En definitiva, cada componente del centroide recoge los valores posibles que toma la misma y la probabilidad con la que lo hace, captando así una mayor información acerca de los individuos que forman los clusters.

Distancia entre individuo y centroide

El cálculo de las distancias entre individuos y centroides es necesario para reubicar al primero en un conglomerado en particular y para redefinir cada uno de los centroides de nuevo en vista a las posibles alteraciones de quienes lo componen.

La distancia entre la j-ésima componente de un individuo X (x_j) y la j-ésima componente de un centroide C_l (c_{lj}) viene determinada por:

$$\delta(x_j,c_{lj}) = \sum_{h=1}^{m_j} p_{ljh}\ \delta(x_j,c_{ljh}),$$

con $\delta(x_j,c_{ljh})$ la distancia entre dos valores categóricos para la j-ésima variable que definen Ahmad y Dey. Tras la definición de la distancia para cada componente, la correspondiente entre individuo y centroide se define como la media de las distancias entre cada una de sus componentes:

$$d(X,C_l) = \tfrac{1}{m}\sum_{j=1}^{m} \delta(x_j,c_{lj}).$$

De la misma forma que las distancias entre individuos eran ponderadas por la correlación existente entre ambos, las correspondientes a individuo y centroide también. De una forma análoga se definen:

$$Corr(X,C_l) = \tfrac{Cov(X,C_l)}{\sqrt{Var(X)}\sqrt{Var(C_l)}},$$

siendo la covarianza:

$$\begin{aligned} Cov(X,C_l) = \sum_{j=1}^{m-1}\sum_{t\ \in T_j}\sum_{t+1\ \in T_{j+1}} & [Div(j,x_j,x_{(j+1)})Div(j,x_j,c_{lj(t+1)})p_{lj(t+1)} \\ & + p_{ljt}Div(j,c_{ljt},x_{(j+1)})Div(j,c_{ljt},c_{lj(t+1)})p_{lj(t+1)}]. \end{aligned}$$

Con respecto a la varianza de un centroide, se determina mediante:

$$Var(C_l) = \sum_{j=1}^{m-1}\sum_{t\ \in T_j} 2p_{ljt}(Div(j,c_{ljt},z_{j+1}))^2,$$

con

$$z_{j+1} = arg \max_{t+1 \in T_{j+1}} Div(j, c_{ljt}, t+1).$$

Con todo ello, la distancia entre el individuo X y el centroide C_l viene determinada a través de la siguiente expresión:

$$d_{corr}(X, C_l) = \Phi_k(Corr(X, C_l))\ d(X, C_l).$$

Tras hallar la distancia entre cada individuo X con respecto de cada uno de los centroides, se asigna el individuo al cluster cuyo centroide esté más cerca del individuo.

Actualización de los centroides

Una vez asignado cada elemento a un cluster, se recalculan los centroides de los mismos.

Iteraciones del proceso

Se repiten los dos pasos anteriores hasta la convergencia del algoritmo. Se entiende por convergencia la estabilización del proceso (en el sentido en el que en dos etapas consecutivas los elementos sigan perteneciento al mismo cluster) o que se sobrepase un número concreto de iteraciones.

Capítulo 2

Simulaciones

La segunda parte de esta memoria está diseñada con el fin de realizar una comparación objetiva del comportamiento para el análisis cluster de secuencias de datos categóricos de algunos de los procedimientos descritos en el capítulo anterior. Dado que el análisis cluster es un proceso de aprendizaje no supervisado, es decir, se desconoce a priori la estructura de grupos subyacente a la base de datos; se efectúa este análisis vía estudios de simulación, donde se generan muestras de datos a partir de escenarios conocidos y preestablecidos de antemano siendo conocida la verdadera estructura de cluster. De este modo se podrán obtener índices de calidad cluster basados en evaluar el grado de acuerdo entre las soluciones experimentales y la solución real. Ésto se hará para diferentes escenarios, manejando distintos parámetros de interés: número de conglomerados, número de etiquetas o atributos categóricos, longitudes de las secuencias, estructura de la dependencia temporal en los distintos clusters, volumen de ruido, . . . Esencialmente, el objetivo no es otro que un proceso de evaluación de los métodos en escenarios simulados que podrían reflejar una situación real en contextos de series con respuesta categórica.
Ciertamente, más escenarios de los aquí presentados pueden estar presentes en situaciones reales, pero los que así se generan son en cualquier caso perfectamente plausibles y permiten enfatizar las ventajas y desventajas que unos métodos reportan respecto de los otros.

El capítulo está estructurado como sigue. En la primera sección se introducen los distintos índices de calidad del ajuste que serán calculados para cada simulación. A continuación se definen los cinco escenarios considerados, justificando su propuesta en base a las características que se desean analizar en cada caso. Tras un breve recordatorio de los algoritmos tratados en el trabajo que serán implementados, se analizan los resultados obtenidos en base a las tablas y gráficos que se muestran a la par que se detallan los resultados o en el apéndice de este documento.

2.1. Índices de calidad de la solución cluster

La calidad de un algoritmo cluster puede ser medida mediante diversos coeficientes que existen en la literatura. Sean $\mathscr{T} = \{\mathscr{T}_1, \ldots, \mathscr{T}_k\}$ el conjunto de cluster reales (conocidos por haber sido simulados) y $\mathscr{E} = \{\mathscr{E}_1, \ldots, \mathscr{E}_s\}$ la configuración de clusters obtenida el método en cuestión ($\mathscr{T}$ y $\mathscr{E}$ pueden no coincidir en su cardinal). Son descritos a continuación algunos de los índices desarrollados en la literatura para evaluar el grado de acuerdo entre $\mathscr{T}$ y $\mathscr{E}$; es decir, el grado de calidad de la solución cluster.

2.1.1. Índice de Gavrilov

Gavrilov et al. (Gavrilov et al. (2000)), proponen un índice basado en la composición de los clusters que se define como:

$$GI(\mathscr{T}, \mathscr{E}) = \frac{1}{k} \sum_{i=1}^{k} \underset{1 \leq j \leq s}{\text{máx}} GI(\mathscr{T}_i, \mathscr{E}_j)$$

donde

$$GI(\mathscr{T}_i, \mathscr{E}_j) = \frac{2|\mathscr{T}_i \bigcap \mathscr{E}_j|}{|\mathscr{T}_i| + |\mathscr{E}_j|}$$

siendo $|\cdot|$ la notación utilizada para cuantificar el número de elementos del conjunto correspondiente. El índice de Gavrilov es un valor numérico perteneciente al intervalo $[0,1]$ dado por un promedio de valores, que suponen la proporción máxima de elementos en común entre un cluster teórico concreto ($\mathscr{T}_i$) con respecto de los generados por el algoritmo ($\mathscr{E}_1, \ldots, \mathscr{E}_s$). Nótese que un valor de *GI* próximo a 0 denota que los clusters obtenidos ($\mathscr{E}$) son muy distintos de los teóricos ($\mathscr{T}$), mientras que valores cercanos a 1 reflejan un mayor acuerdo entre solución teórica y experimental.

2.1.2. Índice Rand

Al igual que el índice de Gavrilov, el índice de Rand (Rand (1971)) toma valores en $[0,1]$. A diferencia del primero se trata de un índice basado en la relación entre individuos. Cuantifica la proporción de pares de elementos que se encuentran bien colocados; entendiendo por ello que si en la solución real se encuentran en un mismo (o distinto) cluster; entonces el algoritmo cluster también los ubica en el mismo (o distinto) conglomerado.
De esta forma el índice se define como:

$$RI = \frac{1}{\binom{n}{2}} \sum_{i<j} Ri(X_i, X_j)$$

con $Ri(X_i, X_j)$ valiendo 1 en caso de que la solución cluster teórica coincida con la experimental generada por el algoritmo (en términos de pertenencia o no a un mismo conglomerado) y 0 en otro caso.

2.1.3. Índice Rand ajustado

El índice Rand ajustado se determina de la siguiente forma:

- Se construye una tabla de la siguiente forma:

$\mathscr{T} \backslash \mathscr{E}$	$\mathscr{E}_1$	$\mathscr{E}_2$	...	$\mathscr{E}_S$	$\sum$
$\mathscr{T}_1$	n_{11}	n_{12}	...	n_{1s}	a_1
$\mathscr{T}_2$	n_{21}	n_{22}	...	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathscr{T}_k$	n_{k1}	n_{k2}	...	n_{ks}	a_k
$\sum$	b_1	b_2	...	b_s	

  donde $n_{ij} = \mid \mathscr{T}_i \bigcap \mathscr{E}_j \mid$ denota el número de individuos incluidos en ambos conjuntos a la vez.

- A partir de los valores de la tabla anterior se define el coeficiente rand ajustado como:

$$IR_{adj} = \frac{\sum\limits_{i,j} \binom{n_{ij}}{2} - \frac{\sum\limits_i \binom{a_i}{2} \sum\limits_j \binom{b_j}{2}}{\binom{n}{2}}}{\frac{1}{2}\left[\sum\limits_i \binom{a_i}{2} + \sum\limits_j \binom{b_j}{2}\right] - \frac{\sum\limits_i \binom{a_i}{2} \sum\limits_j \binom{b_j}{2}}{\binom{n}{2}}}.$$

El índice Rand ajustado puede tomar valores en el intervalo $[-1, 1]$ de forma que: una cifra próxima a 1 determina una asignación correcta de los individuos, un valor cercano a 0 indica una asignación con los mismos resultados que si hubiese sido efectuada al azar y valores negativos colindantes a -1 dan a entender una asignación de individuos muy deficiente con respecto a la real.

2.1.4. Ancho de silueta promedio

Denotado por ASW (*Average Silhouette Width*), este índice viene definido como:

$$ASW = \frac{1}{n} \sum_{i=1}^{n} sil(i)$$

con $sil(i) = \frac{b(i) - a(i)}{\max\{b(i), a(i)\}}$, siendo $a(i)$ la distancia media entre el i-ésimo individuo y el resto de individuos asignados al mismo conglomerado; y $b(i)$ la distancia media entre el individuo i-ésimo y los elementos del cluster más cercano (exceptuando el de pertenencia).

El índice ASW pretende captar lo sólida que resulta la estructura de conglomerados dentro de cada escenario de trabajo. Así, un índice ASW próximo a 1, da a entender que los cluster propuestos por el método en cuestión están bien definidos en un sentido estructural, es decir; los elementos pertenecientes a un mismo conglomerado están muy próximos entre sí y distan bastante del resto de elementos ajenos a su conjunto de asignación.
Nótese que, a diferencia de los índices previos, el ASW se define a partir de la disimilaridad entre objetos, luego sólo será evaluable con procedimeintos de *clustering* que dispongan de esas disimilaridades (no será el caso por ejemplo de algoritmos basados en modelos).

2.2. Escenarios de simulación

A través de los coeficientes mencionados, se medirá la bondad del ajuste efectuado para cada uno de los escenarios simulados definidos a continuación. Cada uno de ellos pretende captar distintas características o circunstancias y analizar cómo se comportan los distintos algoritmos implementados en el lenguaje de programación R.

2.2.1. Escenario 1

En el primero de los escenarios cada individuo viene definido por una secuencia de variables aleatorias que toman valores entre 10 respuestas categóricas posibles (`NCat=10`).
Cada una de estas secuencias que definen a un usuario viene determinada por un número de variables que oscila entre un mínimo de 7 (`MinEnt=7`) y un máximo de 16 (`MaxEnt=16`).

Se consideran 6 clusters con 400 individuos cada uno. Cada uno de estos seis clusters se genera de acuerdo a los siguientes patrones:

- ***Cluster 1***: formado por individuos que únicamente se mueven entre las categorías de la 1 a la 4.
- ***Cluster 2***: formado por individuos que únicamente se mueven entre las categorías de la 5 a la 7.
- ***Cluster 3***: formado por individuos que únicamente se mueven entre las categorías de la 8 a la 10.
- ***Cluster 4***: formado por individuos que comienzan moviéndose por las categorías de la 1 a la 4 (en las primeras cuatro variables) y que posteriormente pasan a hacerlo por las categorías de la 5 a la 7.

- ***Cluster 5***: formado por individuos que comienzan moviéndose por las categorías de la 1 a la 4 (en las primeras cuatro variables) y que posteriormente pasan a hacerlo por las categorías de la 8 a la 10.

- ***Cluster 6***: formado por individuos que comienzan moviéndose por las categorías de la 5 a la 7 (en las primeras cuatro variables) y que posteriormente pasan a hacerlo por las categorías de la 8 a la 10.

Con esta estructura para los clusters, se tiene que los tres primeros son puros en el sentido de que toman valores únicamente dentro de un grupo reducido de posibles categorías. Por otro lado los tres restantes se mueven inicialmente por un grupo reducido, pero posteriormente pasan a otro distinto.
Un ejemplo que motiva un planteamiento como éste del primer escenario puede ser el comportamiento de los usuarios en la navegación de un portal web de noticias. Los tres primeros clusters se corresponderían con usuarios con intereses específicos en un área particular compuesta por cuatro categorías. Así, se podría definir el cluster 1 como el compuesto por los usuarios que navegan a través de las secciones economía, política, nacional e internacional. El cluster 2 aunaría individuos con intereses más culturales: deportes, cine, música y Mundial 2014; mientras que los del cluster 3 presentan intereses más tecnológicos: ciencia, divulgación, nuevas tecnologías e informática. El resto de grupos, lo forman usuarios con los mismos intereses que los tres primeros pero sin una predilección tan específica del tema, por lo que dado un momento cambian completamente de tema para seguir documentándose a cerca de otros asuntos.
Otro ámbito que se ajusta a las características del primer escenario de simulación puede ser el referente al comportamiento de individuos a la hora de realizar la compra en un centro comercial. Si bien algunos pueden estar interesados exclusivamente en productos alimenticios, zona electrónica, ropa, ... otros clientes pueden mostrar un comportamiento más variado y visitar diversas zonas dentro del centro.

Para la generación de los individuos se utiliza una función auxiliar a partir de la cuál se crean una cantidad elevada de usuarios en base a los siguientes parámetros:

- `NCat`: número de categorías que puede tomar cada una de las variables que definen al individuo. Pese a considerar en este caso que todas las variables toman valores de un mismo conjunto de posibilidades, los algoritmos están diseñados para que esta condición no sea estricta y cada una de ellas pueda tener su propio dominio.

- `MinEnt`: número mínimo de variables con respuesta no nula que definen al individuo.

- `MaxEnt`: número máximo de variables con respuesta no nula que definen al individuo.

- `PPaso`: matriz con las probabilidades de transición de una categoría a otra. Su valor es fijo para cualquier variable. Si cada variable tuviese su propio dominio de posibles respuestas, habría que considerar (para usuarios definidos con un máximo de m variables $\{A_1,\ldots,A_m\}$) un total de $m-1$ matrices de transición. Cada una de estas matrices se corresponde con las probabilidades de transición de los estados de la variable i - ésima (A_i) a los estados de la variable $i+1$-ésima (A_{i+1}), luego de dimensión $\mid dom(A_i) \mid \times \mid dom(A_{i+1}) \mid$.

- `P`: vector con las probabilidades iniciales para cada una de las categorías. En este caso particular se considera una distribución uniforme discreta; es decir, la probabilidad de que un individuo comience visitando la i-ésima categoría es de $\frac{1}{\texttt{NCat}} \quad \forall\, i \in \{1,\ \ldots,\ \texttt{NCat}\}$

- `pNA`: probabilidad de obtener un `NA`. Desde el momento en que se tiene un mínimo de `MinEnt` valores para un usuario, se introduce la posibilidad de que se termine la secuencia con una probabilidad de `pNA`. De obtener el valor `NA`, las categorías restantes hasta `MaxEnt` tomarán valores de `NA` igualmente.

- `N`: número de usuarios a simular en base a los parámetros anteriores.

Para este escenario se han simulado 200000 individuos de los cuales luego se trabaja únicamente con 2400 (400 por cada cluster). La selección de los mismos se hace en base a las características que deben cumplir para pertenecer a un conglomerado específico. El hecho de proporcionar una matriz de paso, provoca la existencia de algunos de ellos que no se ajustan a ninguno de los patrones que define un conglomerado.

Con una adecuada selección de los usuarios, se tendría el conjunto total con el que se va a utilizar cada uno de los algoritmos de *clustering* implementados.

2.2.2. Escenario 2

Complicando un poco más las diferencias entre conglomerados, se consideran 3000 individuos definidos por secuencias de variables aleatorias que toman valores en 13 posibles categorías (ésto aumenta el tiempo computacional de los algoritmos de forma considerable respecto del primer escenario). Las secuencias que definen a cada usuario vienen siendo la concatenación de valores para un total de entre 10 y 23 variables aleatorias. Una vez definidas las diez primeras variables, se sigue definiendo el resto hasta llegar a la vigesimotercera teniendo en cuenta que la posibilidad de que el usuario deje de navegar es de 0.05 y que una vez se obtenga un valor de omisión, el resto de variables hasta la última son nulas igualmente.

Bajo estas condiciones, se consideran los cuatro conglomerados siguientes:

- ***Cluster 1***: formado por los usuarios cuya secuencia de 4 primeros valores viene dada por: $1-2-3-4$.
- ***Cluster 2***: formado por los usuarios cuya secuencia de 4 primeros valores viene dada por: $3-5-6-3$.
- ***Cluster 3***: formado por los usuarios cuya secuencia de 4 primeros valores viene dada por: $2-7-2-3$.
- ***Cluster 4***: formado por los usuarios cuya secuencia de 4 primeros valores viene dada por: $1-6-8-7$.

Cabe destacar que los individuos de los conglomerados 1 y 4 comienzan por el mismo valor y que los del 2 y el 3 comparten el cuarto valor de la secuencia. A mayores, para aumentar la dificultad de detección de los conglomerados se han simulado usuarios (un 25 % del total) que no siguen completamente las características de los definidos, pero que sí comparten características en un 75 %.
Así por ejemplo, para el primer conglomerado se han simulado usuarios de la forma $\boxdot-2-3-4$, $1-\boxdot-3-4$, $1-2-\boxdot-4$ y $1-2-3-\boxdot$ donde en lugar de $\boxdot$ se puede tener cualquier valor categórico.
Finalmente, 207 de los 3000 usuarios simulados se corresponden con individuos fuera de estos conglomerados y son considerados como ruido. No se tienen en cuenta de cara a determinar los coeficientes de bondad del proceso, pero sí a lo largo del mismo.

2.2.3. Escenario 3

En la tercera simulación se consideran diversos conglomerados que referencian a usuarios con intereses muy similares (mismos valores de las respuestas categóricas), si bien la estancia en cada una de esas categorías es distinta.
Así, considerando cada variable aleatoria como una unidad de tiempo, la sucesión que define a un usuario dada por una longitud máxima (y mínima en este caso) de 15 respuestas categóricas (`MaxCat` = 15), representaría en el caso de navegación web a las categorías visitadas por el individuo en 15 unidades de tiempo.

En este escenario, los usuarios son individuos con un interés particular en las secciones 1, 2 y 3. No obstante, cada uno de ellos invierte una cantidad distinta de tiempo en cada una de las tres

categorías y finalmente sigue navegando de una forma aleatoria por el resto de las ocho secciones (`NCat` = 8).
Bajo estos patrones, los clusters definidos son los siguientes:

- ***Cluster 1***: el usuario navega por la categoría 1 durante los cuatro primeros intervalos de tiempo, posteriormente pasa otros cuatro instantes en la categoría 2 y tras dos intervalos en la categoría 3 se mueve de forma aleatoria por todas las posibles categorías.

$$1-1-1-1-2-2-2-2-3-3-\boxdot-\boxdot-\boxdot-\boxdot-\boxdot$$

- ***Cluster 2***: un individuo perteneciente a este conglomerado transcurre las seis primeras unidades de tiempo en la categoría 1. Tras tres instantes en la segunda categoría y dos intervalos de tiempo en la categoría 3, comienza a navegar sin un patrón fijo.

$$1-1-1-1-1-1-2-2-2-3-3-\boxdot-\boxdot-\boxdot-\boxdot$$

- ***Cluster 3***: a diferencia de los usuarios de los dos conglomerados anteriores, un individuo del cluster 3 comienza navegando cinco unidades de tiempo en la categoría 2, pasando 4 instantes por la categoría 1 prosigue con un par de intervalos en la categoría 3 para finalizar con una navegación aleatoria.

$$2-2-2-2-2-1-1-1-1-3-3-\boxdot-\boxdot-\boxdot-\boxdot$$

- ***Cluster 4***: un par de instantes de tiempo en las categorías 1, 2 y 3 respectivamente para después navegar de forma aleatoria, definen a los individuos de este conglomerado.

$$1-1-2-2-3-3-\boxdot-\boxdot-\boxdot-\boxdot-\boxdot-\boxdot-\boxdot-\boxdot-\boxdot$$

De cada uno de los escenarios se han generado 500 individuos para realizar el análisis cluster.

2.2.4. Escenario 4

El cuarto de los escenarios trata de clasificar individuos con un comportamiento distinto a los escenarios precedentes. Para cada conglomerado, las entradas segunda, cuarta y sexta serán valores categóricos propios del cluster mientras que el resto de las entradas hasta un total de once (`MaxEnt` = 11) serán cualquier valor de los ocho (`NCat` = 8) considerados como posible respuesta.

Los conglomerados considerados para esta simulación son:

- ***Cluster 1***: las variables segunda, cuarta y sexta quedan fijadas con los valores 2,4 y 6 respectivamente.

$$\boxdot - 2 - \boxdot - 4 - \boxdot - 6 - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

- ***Cluster 2***: en este conglomerado son los valores 1,3 y 5 los que permanecen fijos para las variables segunda, cuarta y sexta respectivamente.

$$\boxdot - 1 - \boxdot - 3 - \boxdot - 5 - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

- ***Cluster 3***: las variables fijas singuen siendo la segunda, la cuarta y la sexta para este conglomerado, siendo 8,7 y 6 sus valores fijos.

$$\boxdot - 8 - \boxdot - 7 - \boxdot - 6 - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

- ***Cluster 4***: a mayores se ha considerado un conglomerado formado por individuos cuyo patrón se corresponde con una trayectoria totalmente aleatoria a través de las 8 categorías posibles, de forma que en cada variable la probabilidad de obtener cualquier respuesta es de $\frac{1}{8}$.
  Su inclusión en la simulación pretende aportar ruido para hacerla más compleja.

$$\boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

En la simulación se han considerado 600 individuos de los primeros tres clusters y 200 del cuarto, si bien a la hora de calcular el valor de los índices, los individuos de este último no han sido considerados.

2.2.5. Escenario 5

El último de los escenarios simulados, resulta similar al cuarto, si bien en esta ocasión se han fijado para cada cluster las entradas segunda y cuarta con diferentes valores de entre los 10 posibles (`NCat` = 10) que definirán cada una de las diez variables que concretan el comportamiento de cada usuario (`MaxEnt` = 10).
La creación de este escenario atiende a las dificultades que surgen a la hora de evaluar las características deseadas si no se concretan de forma adecuada los parámetros para la simulación. Tal y como se comentará más adelante, el número de los parámetros resulta tan importante que de forma involuntaria pueden dar lugar a una correlación que no se pretendía generar.

El quinto escenario está compuesto por los siguientes conglomerados:

- ***Cluster 1***: las variables segunda y quedan fijadas con los valores 1 y 2 respectivamente, dejando abierta la posibilidad de cualquier otra respuesta para las variables restantes.

$$\boxdot - 1 - \boxdot - 2 - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

- ***Cluster 2***: en este conglomerado son los valores 3 y 4 los que ocupan las variables fijas.

$$\boxdot - 3 - \boxdot - 4 - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

- ***Cluster 3***: las respuestas 5 y 6 permanecen invariables para todos los usuarios de este conglomerado como respuesta a las variables segunda y cuarta respectivamente.

$$\boxdot - 5 - \boxdot - 6 - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

- ***Cluster 4***: a mayores se ha considerado un conglomerado formado por individuos cuyo patrón se corresponde con una trayectoria totalmente aleatoria a través de las 10 categorías posibles, de forma que en cada variable la probabilidad de obtener cualquier respuesta es de $\frac{1}{10}$.
  Su inclusión en la simulación pretende aportar ruido para hacerla más compleja.

$$\boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot - \boxdot$$

En la simulación se han considerado 600 individuos de los primeros tres clusters y 200 del cuarto, si bien a la hora de calcular el valor de los índices, los individuos de este último no han sido tomados en cuenta.
En la Tabla 2.1 se han recogido los principales parámetros que definen a cada uno de los escenarios considerados en las simulaciones.

	Ncat	MinEnt	MaxEnt	pNA
Escenario 1	10	7	16	0.15
Escenario 2	11	10	23	0.05
Escenario 3	8	15	15	0
Escenario 4	8	11	11	0
Escenario 5	10	10	10	0

Tabla 2.1: Parámetros para la simulación de los distintos escenarios

Cada situación reflejada mediante un escenario distinto, pretende analizar diversos aspectoos de posibles comportamientos para los usuarios. Más concretamente:

- **Escenario 1**: el primero de los escenarios pretende recrear un ámbito de conglomerados puros (en sus tres primeros clusters) y complicar ligeramente la clasificación con los tres últimos grupos.
  Los clusters puros reflejan grupos de usuarios que se caracterizan por unas preferencias muy específicas que no tienen valores en común con los otros dos existentes. La dificultad surge al introducir otros grupos de usuarios con intereses compartidos entre las categorías incluidas en dos de los clusters puros. Es de esperar que los algoritmos cluster proporcionen buenos resultados en esta situación razonablemente simple.

- **Escenario 2**: es sensiblemente más complejo que el anterior. En primer lugar porque ahora cada cluster está definido por un patrón de conducta en las primeras entradas. No se caracterizan por unas secciones preferentes (de hecho usuarios de distintos clusters comparten secciones visitadas), sino por un orden preestablecido en los accesos de las primeras entradas. La idea es enfatizar el concepto de correlación o dependencia temporal. Cada cluster está determinado por un patrón de conducta temporal en las primeras cuatro entradas. En segundo lugar, se introduce un nivel de ruido importante . Por todo ello es de esperar que algoritmos como el de las k -modas o el de Ahmad y Dey ofrezcan peores resultados que el de García-Magariños y Vilar debido a que el primero no considera una relación temporal entre las variables y el segundo no incluye la correlación en su definición de métrica.

- **Escenario 3**: siguiendo con el ejemplo de los usuarios que navegan en una web de noticias, el gestor del portal podría estar interesado en diferenciar a dos usuarios que visitan, digamos política y economía, pero el primero está prácticamente todo el tiempo en política entrando en economía en el tramo final de navegación, mientras que el segundo dedica el mismo tiempo (y en el mismo orden) a ambas. El tercer escenario busca evaluar la capacidad de los procedimientos clustering examinados para discriminar entre clusters de este tipo. No se trata de analizar cómo de bien detecta cada algoritmo la diferenciación de usuarios por las categorías de interés

sino por la estancia en las mismas. Para ello se considera cada variable como una unidad de tiempo de forma que cuanto mayor sea el número de variables consecutivas con una misma respuesta, más interés denota por parte del usuario en la misma.

- **Escenario 4**: con este escenario se proponen dos objetivos. Primero, generar una situación donde no existe una correlación entre diferencias de primer orden, de modo que cabe esperar que la corrección por correlación que introduce la disimilaridad de García-Magariños y Vilar aporte sólo ruido y deba dársele menos peso en la medida de dismilaridad frente a la propuesta de Ahmad y Dey. Por otro lado, se está generando realmente una situación de dependencia de retardo dos, lo cuál sugiere una posible extensión del procedimiento de García-Magariños y Vilar para cubrir este tipo de correlacines.

- **Escenario 5**: el último escenario es una pequeña variante del cuarto. El número de variables, el número de categorías por cada variable y el número de clusters considerados en la simulación anterior, inducen una correlación de orden uno que impide evaluar de forma correcta la idea expuesta. Con pequeñas variaciones en los parámetros, se logra diluir esa correlación manteniendo la de orden dos fijada por los valores de las variables segunda y cuarta para cada uno de los clusters.

2.3. Algoritmos

La conducta de algunos de los procedimientos de análisis cluster para datos categóricos revisados en esta memoria ha sido evaluada en los escenarios de simulación descritos en la sección anterior. Específicamente, se han considerado los algoritmos partitivos de análisis cluster que siguen.

Algoritmo de las *k*-modas basado en la simple matching dissimilarity (SMD): método de partición basado en disimilaridad siguiendo la propuesta de Huang (Huang (1998)). La matriz de disimilaridades entre pares de individuos se obtiene usando la métrica SMD, de tal forma que, obviamente, el carácter dinámico de las secuencias registradas para cada usuario no es tenido en cuenta.

Algoritmo de Ahmad y Dey: siguiendo el algoritmo de partición propuesto por los autores (Ahmad and Dey (2007)). De nuevo un procedimiento basado en una métrica, que difiere del anterior en la disimilaridad de partida (basada ahora en coocurrencias) y en la determinación de los centroides (distribución de frecuencias para los atributos dentro de cada grupo).

Algoritmo de García-Magariños y Vilar: de acuerdo al procedimiento basado en disimilaridad propuesto por los autores (García-Magariños and Vilar, 2014). La métrica de Ahmad y Dey se pondera en base a una medida de correlación local entre observaciones contiguas, penalizando registros

incorrelados y estableciendo de este modo un concepto de similaridad que tiene en cuenta conductas temporales parejas.

Algoritmo de Pamminger y Frühwirth-Schnatter: procedimiento introducido en Pamminger and Frühwirth-Schnatter (2010). De todos los considerados es el único que realiza la clasificación en base a un modelo prefijado sobre los individuos (modelo de Cadenas de Markov) y sin considerar ningún tipo de disimilaridad

Al respecto de los algoritmos y su ejecución, conviene destacar que:

- las propuestas de Huang, Ahmad & Dey y García-Magariños & Vilar han sido ejecutadas a partir de la programación de estos últimos. Al tratarse de algoritmos basados en una distancia de disimilaridad requieren únicamente determinar el número de clusters que se desea generar.

- el algoritmo de Pamminger y Frühwirth-Schnatter ha sido ejecutado mediante la función `mcClust` que puede encontrarse en la librería `bayesMCClust` del lenguaje de programación R. A diferencia de los otros algoritmos, requiere de diversos parámetros de entrada para su ejecución, si bien en la documentación de las funciones no se especifica cuáles son los más recomendables para cada caso concreto.
  En las simulaciones ejecutadas para el presente trabajo, en lo correspondiente a este algoritmo, se han utilizado los parámetros iniciales por defecto propuestos por los autores, con la salvedad del número de conglomerados a generar.

- los algoritmos de Ahmad & Dey y García-Magariños & Vilar, han sido ejecutados en cada simulación en base a dos criterios. El primero considera una asignación inicial de los individuos a cada conglomerado de forma aleatoria y el segundo una basada en el algoritmo de las k - medias.

2.4. Resultados

La ejecución de los procedimientos *clustering* investigados en los diferentes escenarios de simulación permiten obtener conclusiones acerca de la eficacia en el proceso de clasificación de cada uno de ellos. Más allá, es posible descubrir los beneficios y dificultades que reporta su aplicación en diferentes contextos. A fin de obtener resultados más precisos, los algoritmos han sido ejecutados sobre varias réplicas en cada escenario. En concreto se han generado 50 réplicas en el Escenario 1 y 25 réplicas en cada uno de los restantes.

De cada simulación se han extraído los índices de calidad del *clustering* definidos en la primera sección del capítulo, teniendo en cuenta que para el algoritmo de Pamminger y Frühwirth-Schnatter no se puede concretar el índice ASW (ancho de silueta promedio) debido a la inexistencia de una métrica en la naturaleza del mismo.

En las siguientes secciones se presentarán los resultados de las simulaciones separadamente para cada escenario. Los promedios de los índices alcanzados en las diferentes réplicas de la simulación son registrados en tablas con la siguiente notación: cada columna indica el nombre de los algoritmos (K-modas, A-D (Ahmad y Dey), GM-V (García-Magariños y Vilar) y Pamminger) donde cuando Asig. alea. se referencia significa asignación aleatoria y Asig. K asignación basada en *k*-medias. En otras tablas se puede ver el número del cluster (C1, C2, ...) o la combinación de dos de ellos (C1-C2, C3-C4, ...). En las tablas que miden las distancias entre individuos de uno mismo cluster o de elementos pertenecientes a dos distintos, se ha añadido una columna final que contabiliza el promedio de los valores referentes a esa fila.
Con respecto a los índices, la notación es GI para el índice de Gavrilov, RI para el índice de Rand, ARI para el índice de Rand ajustado y ASW referente al ancho de silueta promedio. La distribución de estos índices se ilustra mediante diagramas de caja múltiples que permiten asimismo comparar visualmente su comportamiento con los distintos procedimientos.

Para los tres algoritmos basados en disimilaridad, se han generado diagramas de caja con los correspondientes valores entre individuos ubicados en un mismo cluster e individuos ubicados en clusters diferentes. El objetivo de esta representación es apreciar diferencias entre los distintos métodos de forma que si uno ofrece distancias más cortas entre los elementos de un conglomerado, da lugar a un grupo homogéneo (propiedad de interés para los clusters), mientras que si un método determina distancias elevadas entre elementos de distintos clusters, da lugar a heterogeneidad entre los mismos (característica que se desea encontrar). Los términos *distancias cortas* y *distancias elevadas* hacen referencia al valor relativo de las obtenidas entre conglomerados y dentro de un cluster en particular.

Los diagramas de cajas de las distancias, son los correspondientes a un escenario en particular, si bien se ha seleccionado un caso de la simulación representativo del conjunto de todas ellas.

2.4.1. Escenario 1

La Tabla 2.2 permite concluir que el procedimiento de García-Magariños y Vilar conduce a las mejores tasas de clasificación con la totalidad de índices analizados. Cuando este algoritmo parte de una asignación inicial basada en k-medias se alcanzan índices promedio iguales a 1 que indican que las soluciones cluster experimentales coinciden con la solución real. Los resultados empeoran algo cuando se usa la asignación aleatoria, pero moviéndose siempre por encima de 0.8. Los niveles próximos a 1 del ancho silueta promedio demuestran por otro lado la elevada compacidad de los grupos resultantes y por tanto una fuerte estabilidad de las soluciones encontradas. El procedimiento de Ahmad y Dey es el que sigue en el ranking pero con resultados sensiblemente peores. Con este procedimiento no se perciben diferencias sustanciales según el método de asignación inicial empleado y sus mejores índices (IR=0.75 e IG=0.60) están 0.20 por debajo de los alcanzados con el procedimiento de García-Magariños y Vilar. También el silueta promedio, alrededor de 0.45, empeora sustancialmente aunque sí es capaz de delatar una estructura cluster subyacente. En definitiva, el algoritmo Ahmad y Dey trabaja razonablemente bien al beneficiarse de la estructura de clusters puros diseñada para este escenario, donde las coocurrencias de etiquetas tienen lugar en grupos diferentes. Sin embargo, el algoritmo de García-Magariños y Vilar se aprovecha igualmente de esta característica y, por ende, de la estructura de correlación entre etiquetas sucesivas que permite enfatizar las entradas cronológicas donde esas coocurrencias tienen lugar. Esta estructura tan particular es también la que justifica que el algoritmo de Huang basado en k-modas conduzca a resultados próximos a los alcanzados con el procedimiento de Ahmad y Dey, aunque con un silueta promedio próximo a cero que delata una débil compacidad de la solución generada. El método de Pamminger presenta en cambio los peores resultados, lo cual sólo puede ser achacable a la necesidad de ajustar más adecuadamente los parámetros de entrada del algoritmo.

	K-modas	A-D Asig. alea.	A-D Asig. K	GM-V Asig. alea.	GM-V Asig. K	Pamminger
GI	0.49290453	0.6037406	0.5931126	0.8567924	1.0000000	0.4238092
RI	0.78472359	0.7564013	0.7414306	0.9417709	1.0000000	0.4961154
ARI	0.26214064	0.4411270	0.4264955	0.8057155	1.0000000	0.1922208
ASW	0.06642407	0.4453204	0.4735932	0.8342677	0.9736849	-

Tabla 2.2: Escenario 1: promedios sobre las 50 réplicas de la los índices de acuerdo entre soluciones experimentales y solución real.

	C1	C2	C3	C4	C5	C6	Promedio
GM-V							
Mediana	0.00206	0.00170	0.00176	0.00216	0.00266	0.00180	0.00206
Media	0.00344	0.00282	0.00355	0.00320	0.00323	0.00288	0.00319
SD	0.00395	0.00313	0.00430	0.00300	0.00226	0.00300	0.00335
A-D							
Mediana	0.02175	0.01827	0.02009	0.01875	0.01758	0.01812	0.01897
Media	0.02156	0.01785	0.01967	0.01845	0.01731	0.01783	0.01878
SD	0.00577	0.00561	0.00611	0.00466	0.00428	0.00483	0.00545
SMD							
Mediana	0.75000	0.69231	0.69231	0.70000	0.70000	0.66667	0.71429
Media	0.75040	0.66669	0.66716	0.70599	0.70383	0.66699	0.69351
SD	0.16569	0.20028	0.19836	0.16196	0.16150	0.17336	0.18022

Tabla 2.3: Escenario 1: promedio de las distancias entre usuarios pertenecientes al mismo conglomerado.

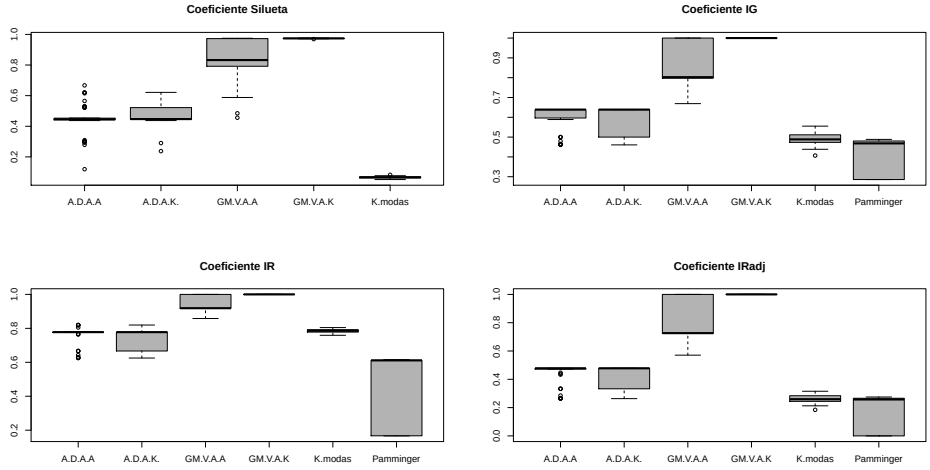

Figura 2.1: Escenario 1: boxplot para los índices de acuerdo entre soluciones experimentales y solución real.

2.4.2. Escenario 2

La Tabla 2.4 refleja la conveniencia de considerar un número de clusters superior al real. Ésto puede ser debido a la inclusión de ruido en la simulación. Los individuos considerados como tal, no son asignables a ningún conglomerado con lo que permitiendo mayor libertad mediante un número mayor de clusters, los resultados muestran una mejoría pese a que el ruido no intervenga en ningún caso a la hora de calcular el valor de los índices de ajuste. Un análisis más detallado de los resultados muestra que aún considerando ocho grupos, algunos de ellos están vacíos o constan únicamente de un elemento. No obstante, esta situación da lugar a un aumento del valor de los índices que determinan la calidad del resultado.

La Tabla 2.4 junto con la Figura 2.2 muestran una mayor compacidad de los grupos formados por los métodos que consideran la correlación como característica a tener en cuenta. De hecho esa diferencia en el ancho silueta promedio no baja de las 0.3 unidades de diferencia, llegando a más de 0.5 en el caso que mejores resultados ofrece el método de García-Magariños y Vilar. La consideración de correlación en el modelo, además de ofrecer una mayor estabilidad en las soluciones, da lugar a una clasificación correcta de los individuos. Los resultados obtenidos para los índices oscilan cerca del valor 1 con mucha menos variabilidad con la que lo hacen el resto de métodos, constatando una buena adaptación del algoritmo propuesto en García-Magariños and Vilar (2014) para las características presentadas por los clusters en esta ocasión.

Con respecto a la elección entre asignación inicial aleatoria o basada en k - medias para los centroides de los conglomerados, los resultados muestran cifras similares que no conducen a ningún tipo de preferencia en la elección.

Un análisis de la Tabla 2.5 y la Tabla 2.6 permite comprender que los resultados ofrecidos por el algoritmo de García-Magariños y Vilar sean mejores que los de Ahmad y Dey. La relación de las distancias entre clusters distintos con respecto a las distancias entre individuos del mismo conglomerado muestra un cambio de valores mayor en el caso con la disimilaridad basada en correlación, resultando así la clasificación más sencilla con este algoritmo. Este hecho junto con que el ancho silueta resulte próximo a 1, evidenciando así compacidad a la hora de formar los conglomerados, da lugar a que el algoritmo propuesto en García-Magariños and Vilar (2014) sea muy competitivo en este tipo de escenarios.

	K-modas	A-D Asig. alea.	A-D Asig. K	GM-V Asig. alea.	GM-V Asig. K	Pamminger
$k = 4$						
GI	0.7571933	0.9067753	0.8669777	0.8916954	0.7501438	0.8309003
RI	0.8411094	0.9285299	0.8995208	0.9175750	0.8076697	0.8629533
ARI	0.5863137	0.8433385	0.7749742	0.8113573	0.5989739	0.6473599
ASW	0.1051555	0.2734774	0.2692708	0.7630921	0.6066104	-
$k = 5$						
GI	0.74053361	0.9580089	0.8818108	0.9582680	0.8166046	0.8109567
RI	0.84327456	0.9672393	0.9092280	0.9679380	0.8581489	0.8583683
ARI	0.56932770	0.9224229	0.7964891	0.9262384	0.6937345	0.6172378
ASW	0.08636343	0.2650083	0.2562229	0.8454413	0.6368464	-
$k = 6$						
GI	0.65591351	0.9110000	0.8519472	0.9616328	0.9313729	0.7802729
RI	0.81604476	0.9316019	0.8849067	0.9724218	0.9479820	0.8445788
ARI	0.47333084	0.8460702	0.7402641	0.9328680	0.8807110	0.5730833
ASW	0.06324072	0.2465977	0.2554304	0.8435082	0.7907795	-
$k = 7$						
GI	0.70383382	0.8679990	0.8319353	0.9894648	0.9803404	0.7840083
RI	0.84112229	0.8927589	0.8600663	0.9918581	0.9851889	0.8446247
ARI	0.52981752	0.7707341	0.7571157	0.9777881	0.9643520	0.5705510
ASW	0.07211788	0.2473775	0.2738419	0.8798414	0.8882376	-
$k = 8$						
GI	0.70175547	0.8627012	0.8569815	0.9773566	0.9917439	0.7645066
RI	0.84522792	0.8853549	0.8914943	0.9857644	0.9939419	0.8379069
ARI	0.52805856	0.7604593	0.7624812	0.9605078	0.9834935	0.5515199
ASW	0.06328019	0.2316916	0.2554338	0.8352110	0.8935718	-

Tabla 2.4: Escenario 2: promedios sobre las 25 réplicas de la los índices de acuerdo entre soluciones experimentales y solución real.

	C1	C2	C3	C4	Promedio
G-M & V					
Mediana	0.00024	0.01675	0.00025	0.02472	0.01765
Media	0.01204	0.01142	0.01145	0.02210	0.01431
SD	0.01508	0.01128	0.01520	0.01078	0.01390
A& D					
Mediana	0.01147	0.01268	0.01137	0.01946	0.01414
Media	0.01300	0.01214	0.01298	0.01952	0.01443
SD	0.00463	0.00286	0.00468	0.00424	0.00511
SMD					
Mediana	0.75000	0.78261	0.75000	0.90476	0.81250
Media	0.76950	0.76955	0.76690	0.87959	0.79712
SD	0.15480	0.12744	0.15008	0.10197	0.14267

Tabla 2.5: Escenario 2: promedio de las distancias entre usuarios pertenecientes al mismo conglomerado.

	C1-C2	C1-C3	C1-C4	C2-C3	C2-C4	C3-C4	Promedio
GM-V							
Mediana	0.02990	0.02260	0.02492	0.03039	0.02695	0.02481	0.02775
Media	0.02623	0.02524	0.02183	0.02711	0.02224	0.02190	0.02409
SD	0.01087	0.00559	0.01237	0.01046	0.01210	0.01246	0.01117
A-D							
Mediana	0.01794	0.01494	0.01811	0.01798	0.01834	0.01809	0.01778
Media	0.01708	0.01588	0.01795	0.01720	0.01819	0.01789	0.01738
SD	0.00356	0.00265	0.00373	0.00341	0.00373	0.00382	0.00360
SMD							
Mediana	0.91304	0.90476	0.90000	0.91304	0.90909	0.90000	0.90909
Media	0.89081	0.89161	0.87038	0.89461	0.87722	0.86790	0.88208
SD	0.11023	0.07061	0.10696	0.10771	0.10761	0.11005	0.10406

Tabla 2.6: Escenario 2: promedio de las distancias entre usuarios pertenecientes a distintos conglomerados.

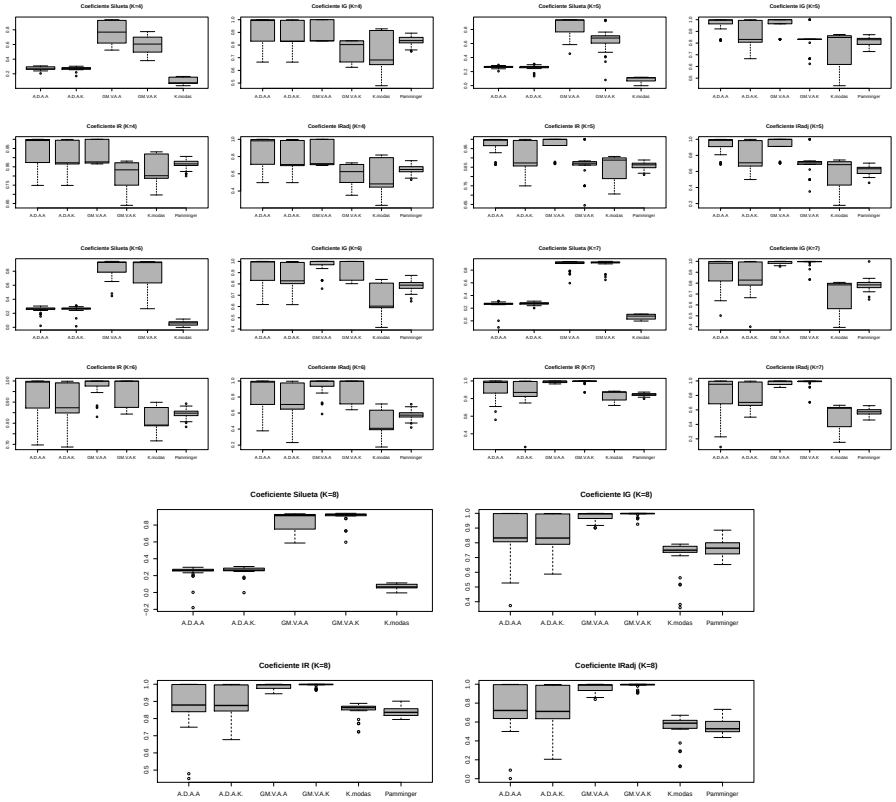

Figura 2.2: Escenario 2: boxplot para los índices de acuerdo entre soluciones experimentales y solución real.

2.4.3. Escenario 3

En el tercer escenario vuelve a resultar como mejor propuesta el algoritmo de García-Magariños y Vilar. El ancho silueta promedio (ASW) vuelve a determinar el algoritmo que considera la correlación entre individuos como el más sólido en cuanto a la formación de conglomerados (ver Tabla 2.7 y Figura 2.3) superando en más de tres décimas los resultados de otros algoritmos. Al igual que en el primer escenario, la asignación inicial basada en el método de las k - medias da lugar a una mejor agrupación de los individuos. En este caso, el algoritmo de García - Magariños y Vilar es el que se ve más beneficiado por esta inicialización del algoritmo.

Es destacable en las simulaciones el comportamiento del algoritmo de las k-modas, ofreciendo mejores resultados que el algoritmo de Ahmad y Dey por ejemplo. Atendiendo a la naturaleza de los datos, resulta comprensible esta situación. Los clusters 1, 2 y 3 están compuestos por individuos que comparten cuanto menos 10, 11 y 11 variables con la misma respuesta respectivamente. Esta situación añadida al hecho de que los usuarios vienen definidos a través de la concatenación de quince valores, da lugar a una distancia máxima de 0.33, 0.27 y 0.27 unidades respectivamente entre elementos pertenecientes al mismo conglomerado (como se refleja en la Tabla 2.8). A poco que la estructura fija de los individuos difiera ligeramente entre clusters, la SMD como métrica determina mayor distancia entre aquellos que pertenecen a distintos conglomerados, ayudando así al algoritmo de las k-modas a realizar una mejor clasificación de éstos.

Un vistazo a las Tablas 2.8 y 2.9 refleja junto con los boxplot de las Figuras A.5 y A.6, mayor distancia entre los distintos conglomerados a la vez que pequeños valores para las dadas entre individuos del mismo cluster en el caso del algoritmo de correlación. Esta diferencia permite comprender los mejores resultados ofrecidos por el algoritmo, si bien cabe destacar que vista la Figura A.6, puede presentar mayores dificultades la decisión de asignar un individuo a un cluster cuando entre el primero y el tercero se trata. Las distancias de la SMD representadas en las Figuras 2.8 y 2.9 muestran en comparación con otros escenarios de simulación unas diferencias más perceptibles entre distancias dentro de un mismo cluster y distancias entre grupos; característica clave en la mejora de los resultados ofrecidos por el algoritmo de las k - modas en relación a otros escenarios.

	K-modas	A-D Asig. alea.	A-D Asig. K	GM-V Asig. alea.	GM-V Asig. K	Pamminger
GI	0.8192351	0.41928235	0.5871551	0.8290018	0.9985184	0.3999984
RI	0.8828184	0.28407886	0.6003166	0.8640448	0.9985446	0.2496448
ARI	0.7136144	0.03218667	0.3439867	0.7132447	0.9961160	0.0000000
ASW	0.3692513	0.58529801	0.5911070	0.9280065	0.9658866	-

Tabla 2.7: Escenario 3: promedios sobre las 25 réplicas de la los índices de acuerdo entre soluciones experimentales y solución real.

	C1	C2	C3	C4	Promedio
GM-V					
Mediana	0.00007	0.00003	0.00002	0.01012	0.00004
Media	0.00013	0.00003	0.00002	0.01365	0.00345
SD	0.00013	0.00001	0.00001	0.01123	0.00814
A-D					
Mediana	0.00154	0.00094	0.00096	0.02910	0.00126
Media	0.00305	0.00095	0.00096	0.03076	0.00893
SD	0.00300	0.00028	0.00029	0.01638	0.01513
SMD					
Mediana	0.33333	0.26667	0.26667	0.53333	0.26667
Media	0.29189	0.23343	0.23341	0.52474	0.32087
SD	0.04918	0.04405	0.04417	0.06614	0.13075

Tabla 2.8: Escenario 3: promedio de las distancias entre usuarios pertenecientes al mismo conglomerado.

	C1-C2	C1-C3	C1-C4	C2-C3	C2-C4	C3-C4	Promedio
GM-V							
Mediana	0.00597	0.25880	0.19971	0.23847	0.21862	0.20084	0.21604
Media	0.00583	0.25711	0.19683	0.23844	0.21841	0.18909	0.18428
SD	0.00059	0.00464	0.02694	0.00070	0.02852	0.03202	0.08565
A-D							
Mediana	0.03066	0.13965	0.12733	0.12676	0.13643	0.14987	0.12786
Media	0.02970	0.13868	0.12117	0.12677	0.13312	0.14340	0.11547
SD	0.00258	0.00258	0.01218	0.00029	0.01227	0.01233	0.04002
SMD							
Mediana	0.53333	0.86667	0.80000	0.80000	0.80000	0.80000	0.80000
Media	0.49083	0.82419	0.79275	0.76671	0.79198	0.79288	0.74322
SD	0.04968	0.04975	0.06567	0.04415	0.06539	0.06714	0.12787

Tabla 2.9: Escenario 3: promedio de las distancias entre usuarios pertenecientes a distintos conglomerados.

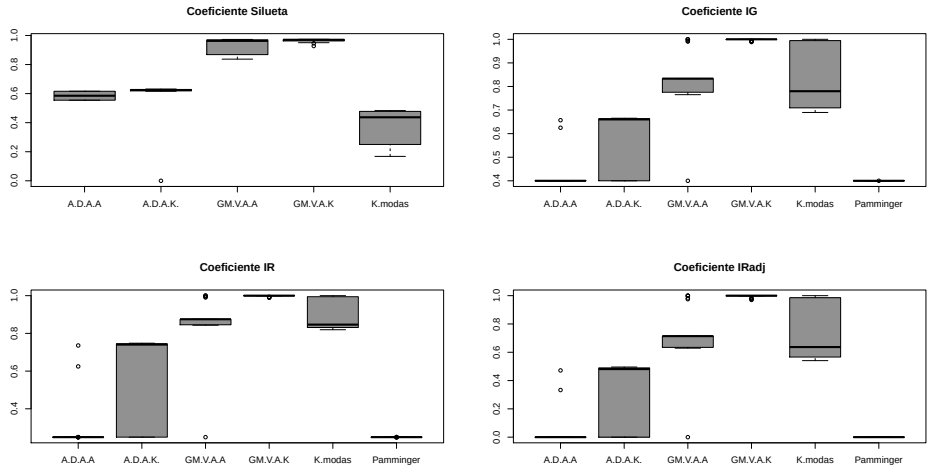

Figura 2.3: Escenario 3: boxplot para los índices de acuerdo entre soluciones experimentales y solución real.

2.4.4. Escenarios 4 y 5

Los dos últimos escenarios de simulación presentados comparten una relación estrecha en cuanto a la formación de los individuos, si bien difieren en parámetros como el número de categorías que definen a cada usuario y el número de respuestas que puede tomar cada una de las variables. Como se comentará a continuación, pequeñas variaciones en estos parámetros pueden generar resultados muy diferentes para cada uno de los métodos.

Una comparativa entre la Tabla 2.10 y la Tabla 2.13 refleja un descenso de los índices de calidad del *clustering* de más de tres décimas en el caso del método de García-Magariños y Vilar, y una mejora sustancial del algoritmo de Ahmad y Dey en el quinto escenario llegando a clasificar de forma correcta todos los usuarios. Con respecto a los algoritmos de k - modas y el de Pamminger, los resultados ofrecidos son similares en los dos escenarios; no obstante la propuesta de Huang (Huang (1998)) muestra unos valores más que aceptables (rondando el 0.85 en índices como el de Gavrilov y el de Rand) mientras que el de Pamminger no llega a sobrepasar el 0.6.

La justificación de estos resultados vuelve a encontrarse en la naturaleza de los individuos que forman los clusters. Limitar a 8 las posibles respuestas para cada variable junto con la simulación de 600 individuos por grupo, da lugar a que se genere correlación entre usuarios (una correlación de orden uno a mayores de la de orden dos que se garantiza por la definición de los individuos). Esta característica beneficia al algoritmo de García-Magariños y Vilar , que con el aumento de posibles respuestas categóricas del quinto escenario se ve superado por el algoritmo de Ahmad y Dey ante la ausencia de correlación de retardo uno.

El algoritmo de k - modas es el que ofrece unos resultados más sorprendentes, si bien puede verse ayudado por el escaso número de variables que definen al usuario en comparación con las respuestas categóricas que se pueden dar para cada una de ellas. Común a todos los algoritmos resulta el ancho silueta promedio bajo que ofrecen todos ellos, dando lugar a la formación de clusters poco compactos.

Contemplando las tablas referentes a las distancias para elementos de un mismo conglomerado y entre individuos pertenecientes a dos distintos (Tabla 2.11, Tabla 2.14, Tabla 2.12 y Tabla 2.15), conviene destacar la similitud entre los resultados para los algoritmos basados en la distancia propuesta por Ahmad y Dey sin correlación y con correlación, si bien estos últimos presentan mayor variabilidad en todas las distancias consideradas. La diferencia de resultados puede ser debida a esa variabilidad que genera mayor confusión a la hora de seleccionar el cluster de asignación de cada elemento para el algoritmo de García-Magariños y Vilar.

Este descenso de la calidad del *clustering* en el algoritmo correlativo, da lugar a una de las posibles líneas de investigación mencionadas posteriormente y que hace referencia a una captación de correlaciones de órdenes mayores para que los resultados ofrecidos sean tan buenos como los del resto de escenarios planteados.

	K-modas	A-D Asig. alea.	A-D Asig. K	GM-V Asig. alea.	GM-V Asig. K	Pamminger
GI	0.8625846	0.8266000	0.6666667	0.8230309	0.8004543	0.5714113
RI	0.8595263	0.8131409	0.5997777	0.8366469	0.8087983	0.6272065
ARI	0.6862572	0.6624940	0.3426937	0.6360560	0.6034852	0.1782827
ASW	0.1620996	0.6562487	0.7043077	0.3498269	0.3680804	-

Tabla 2.10: Escenario 4: promedios sobre las 25 réplicas de la los índices de acuerdo entre soluciones experimentales y solución real.

	C1	C2	C3	Promedio
GM-V				
Mediana	0.01324	0.01317	0.01332	0.01324
Media	0.01309	0.01301	0.01316	0.01309
SD	0.00181	0.00178	0.00179	0.00180
A-D				
Mediana	0.01336	0.01330	0.01344	0.01337
Media	0.01321	0.01315	0.01328	0.01321
SD	0.00168	0.00166	0.00167	0.00167
SMD				
Mediana	0.72727	0.72727	0.72727	0.72727
Media	0.67098	0.67137	0.67148	0.67128
SD	0.06875	0.06835	0.06837	0.06849

Tabla 2.11: Escenario 4: promedio de las distancias entre usuarios pertenecientes al mismo conglomerado.

	C1-C2	C1-C3	C2-C3	Promedio
GM-V				
Mediana	0.03183	0.03304	0.03133	0.03204
Media	0.03169	0.03288	0.03119	0.03192
SD	0.00220	0.00219	0.00220	0.00231
A-D				
Mediana	0.03163	0.03292	0.03115	0.03185
Media	0.03148	0.03276	0.03100	0.03175
SD	0.00168	0.00168	0.00167	0.00183
SMD				
Mediana	1.00000	1.00000	1.00000	1.00000
Media	0.94384	0.94427	0.94414	0.94408
SD	0.06864	0.06834	0.06851	0.06850

Tabla 2.12: Escenario 4: promedio de las distancias entre usuarios pertenecientes a distintos conglomerados.

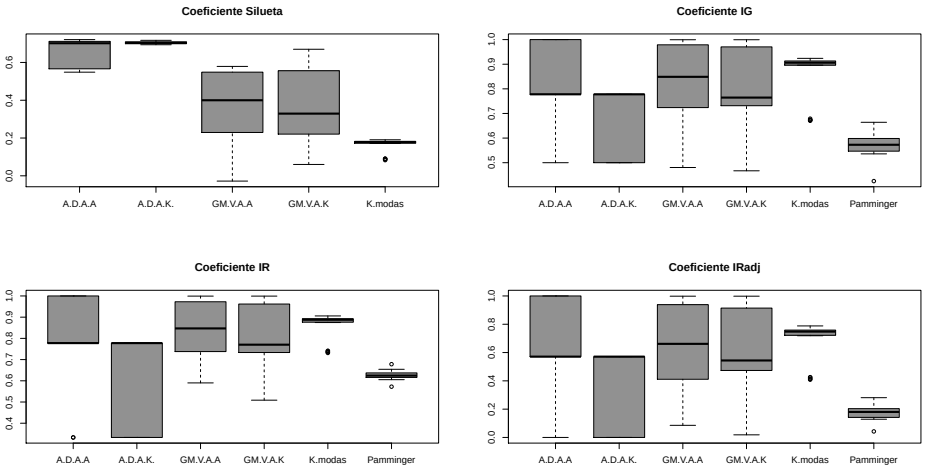

Figura 2.4: Escenario 4: boxplot para los índices de acuerdo entre soluciones experimentales y solución real.

	K-modas	A-D Asig. alea.	A-D Asig. K	GM-V Asig. alea.	GM-V Asig. K	Pamminger
GI	0.8344634	1.0000000	0.9733333	0.51692183	0.53214453	0.45616825
RI	0.8191234	1.0000000	0.9733185	0.61183084	0.62025755	0.57005554
ARI	0.5969941	1.0000000	0.9485387	0.15100796	0.16448961	0.05072759
ASW	0.1203599	0.3842571	0.3697928	0.04970404	0.05701718	-

Tabla 2.13: Escenario 5: promedios sobre las 25 réplicas de la los índices de acuerdo entre soluciones experimentales y solución real.

	C1	C2	C3	Promedio
GM-V				
Mediana	0.00891	0.00883	0.00881	0.00885
Media	0.00880	0.00873	0.00871	0.00875
SD	0.00150	0.00145	0.00147	0.00147
A-D				
Mediana	0.00900	0.00894	0.00892	0.00896
Media	0.00889	0.00884	0.00882	0.00885
SD	0.00124	0.00122	0.00123	0.00123
SMD				
Mediana	0.70000	0.70000	0.70000	0.70000
Media	0.71996	0.71988	0.71997	0.71994
SD	0.08500	0.08482	0.08497	0.08493

Tabla 2.14: Escenario 5: promedio de las distancias entre usuarios pertenecientes al mismo conglomerado.

	C1-C2	C1-C3	C2-C3	Promedio
GM-V				
Mediana	0.01518	0.01478	0.01512	0.01503
Media	0.01506	0.01467	0.01502	0.01492
SD	0.00176	0.00175	0.00173	0.00175
A-D				
Mediana	0.01502	0.01467	0.01503	0.01491
Media	0.01492	0.01456	0.01493	0.01480
SD	0.00123	0.00123	0.00122	0.00124
SMD				
Mediana	0.90000	0.90000	0.90000	0.90000
Media	0.92013	0.91995	0.92000	0.92002
SD	0.08473	0.08474	0.08467	0.08471

Tabla 2.15: Escenario 5: promedio de las distancias entre usuarios pertenecientes a distintos conglomerados.

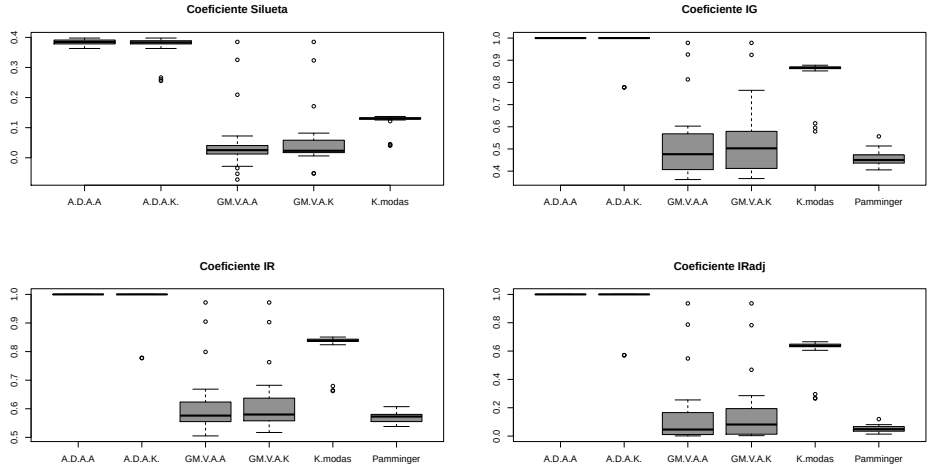

Figura 2.5: Escenario 5: boxplot para los índices de acuerdo entre soluciones experimentales y solución real.

Capítulo 3

Conclusiones y lineas de trabajo futuras

La presente memoria se ha centrado en el problema de análisis cluster de secuencias ordenadas en el tiempo de datos categóricos. Se han descrito algunos procedimientos partitivos propuestos en la literatura, comenzando con las propuestas basadas en k-modas, que no consideran el carácter dinámico de los datos, y siguiendo con técnicas más recientes diseñadas ya para tener en cuenta esta característica. Con la idea de evaluar el comportamiento de estos algoritmos, se ha examinado su comportamiento en distintos escenarios de simulación. A continuación se procede a concretar los aspectos positivos y las limitaciones de los algoritmos tratados junto con alguna idea de cuáles pueden ser las líneas de investigación futuras de cara a mejorar los resultados ofrecidos.

El algoritmo de las k - modas introducido por Huang se basa en medir la disimilaridad entre individuos usando la SMD (*simple matching dissimilarity*) y aplicar la filosofía del k-medias como método iterativo de clasificación, considerando las modas de los grupos como centroides. La SMD no permite captar más que diferencias muy pronunciadas. En los diferentes planteamientos de simulación se puede apreciar cómo ofrece resultados discretos en comparación con el resto de algoritmos.
En cuanto se incluye una estructura de serie de tiempo, el algoritmo de las k - modas presenta muchas dificultades de cara a realizar una clasificación correcta de los individuos. Así pues, su utilización no es apropiada en problemas como el análisis cluster de una base de datos compuesta por el itinerario de usuarios en una página web.

La propuesta de modelizar los conglomerados a través de cadenas de Markov (Pamminger and Frühwirth-Schnatter, 2010) pese a captar la dependencia temporal entre variables, presenta una desventaja con respecto del resto de algoritmos; la necesidad de concretar varios parámetros para poder proceder con su ejecución. En Pamminger (2012) se establecen los parámetros de entrada pero en ningún caso se concreta cómo ajustarlos en función de la base de datos con la que se trabaja. Este hándicap propio de los algoritmos basados en un modelo, se puede apreciar en los resultados que ofrecen las distintas simulaciones; donde se han utilizado los parámetros que vienen por defecto.
De gran interés resultaría construir una metodología que permita estimar los parámetros necesarios

únicamente a partir de la base de datos a analizar. Así, se podrían mejorar unos resultados que mediante los parámetros iniciales sugeridos en Pamminger (2012) ofrecen valores alejados de otros métodos como el de Ahmad and Dey (2007) o el planteado por García-Magariños and Vilar (2014).

La propuesta más reciente de analizar las coocurrencias de sucesos y definir a partir de éstas la distancia entre individuos (Ahmad and Dey, 2007) resulta un algoritmo eficiente en casi todos los escenarios propuestos. Los índices de bondad de ajuste resultan ciertamente aceptables debido en gran parte a la consideración del carácter dinámico de la secuencia de variables mediante la distancia propuesta. No obstante, cuando existe correlación de primer orden en los individuos, la idea de García-Magariños y Vilar que incluye el factor correlación dentro de la definición de distancia (García-Magariños and Vilar, 2014) capta mejor las propiedades del conjunto de datos. Únicamente cuando se modifica la correlación a órdenes mayores resulta ofrecer peores resultados, pues el algoritmo está diseñado para captar esa correspondencia de orden uno. Una de las posibles líneas de investigación es la consideración de órdenes mayores, lo que llevaría a la implementación de algoritmos más complejos y costosos computacionalmente.
Otro de los temas de interés con lo que al algoritmo propuesto en García-Magariños and Vilar (2014) respecta es la función reguladora de la distancia de Ahmad y Dey y la correlación propuesta por García-Magariños y Vilar (ver expresión 1.2). A lo largo de las simulaciones y debido a la sugerencia del artículo, se fija el parámetro k de dicha función con el valor 5. Una modificación del valor de k en base al conjunto de datos con el que se trabaje podría mejorar los resultados obtenidos, pues no todas las bases de datos están formadas por usuarios con variables correladas de la misma manera.
Una tercera vía de interés para el algoritmo de García-Magariños y Vilar es la extensión de éste para datos mixtos. Para ello, se podría tomar la idea de Ahmad y Dey modificándola mediante una correlación no sólo para datos categóricos sino para los datos numéricos que definan al usuario. Conviene destacar también que en Ahmad and Dey (2007) no se especifica cómo dividir los datos numéricos en intervalos; paso que debe de hacerse para calcular las distancias entre valores categóricos de una misma variable y para determinar la significación de las variables numéricas.

Con un aspecto más global y común a todos los algoritmos, se encuentra la incógnita de cuántos conglomerados son convenientes considerar en cada caso. Este aspecto concierne a todos los métodos para el análisis cluster y podría resultar de gran utilidad el hecho de tener una aproximación del número ideal de conglomerados. De obtener una propuesta, en lugar de ejecutar múltiples escenarios para ver los resultados, bastaría enfocar el número de clusters a un entorno de la sugerencia indicada por el método. Tal y como se ha comprobado en las simulaciones, el enfoque debe de ser dirigido a valores ligeramente superiores a la propuesta, donde los algoritmos presentan mejores resultados aún con algunos clusters vacíos o con muy pocos individuos.

En definitiva, el análisis cluster para series de tiempo con respuesta categórica es un área con un margen de mejora importante y a la espera de algoritmos que se adecúen a un mayor rango de escenarios posibles. La gran utilidad de algoritmos de este tipo en diversos campos, tal y como se ha podido ver por ejemplo con los usuarios de una página web, hace aún más interesante la búsqueda de modelos o técnicas de clustering con el fin de comprender los distintos patrones que rigen el comportamiento de los usuarios y utilizar esta información con diferentes fines.

Apéndice

Apéndice A

Gráficos de las simulaciones

A continuación se muestran los gráficos referentes a distancias para elementos dentro de un mismo cluster y aquellas resultantes entre elementos de dos conglomerados distintos.

- **Escenario 1**

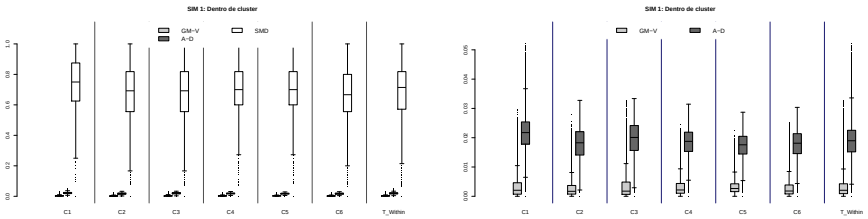

Figura A.1: Escenario 1: boxplot de distancias dentro de cada conglomerado.

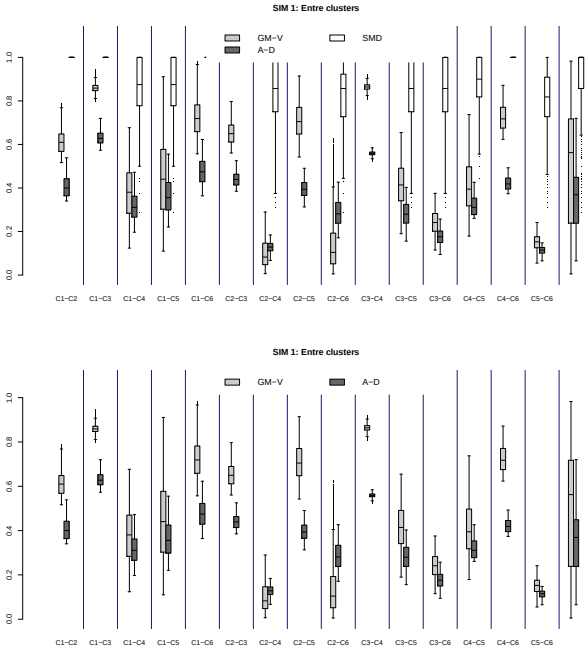

Figura A.2: Escenario 1: boxplot de distancias entre distintos conglomerados.

- **Escenario 2**

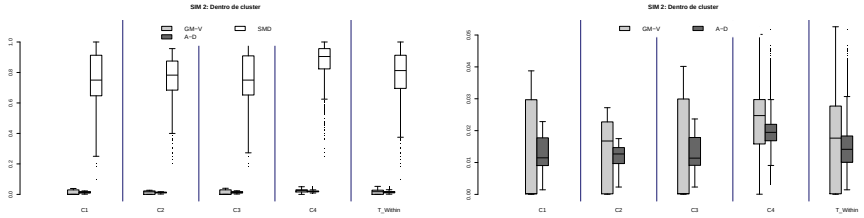

Figura A.3: Escenario 2: boxplot de distancias dentro de cada conglomerado.

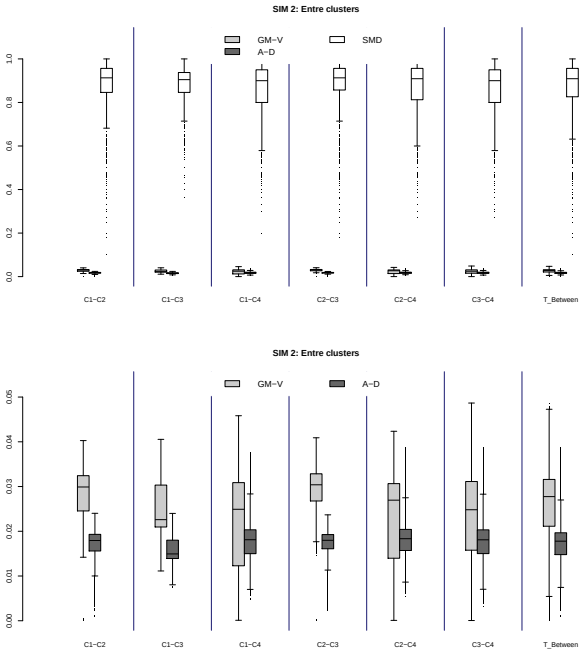

Figura A.4: Escenario 2: boxplot de distancias entre distintos conglomerados.

- **Escenario 3**

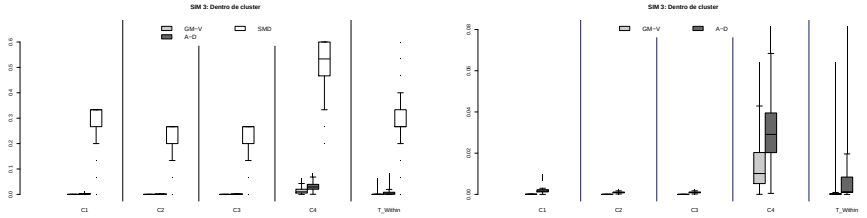

Figura A.5: Escenario 3: boxplot de distancias dentro de cada conglomerado.

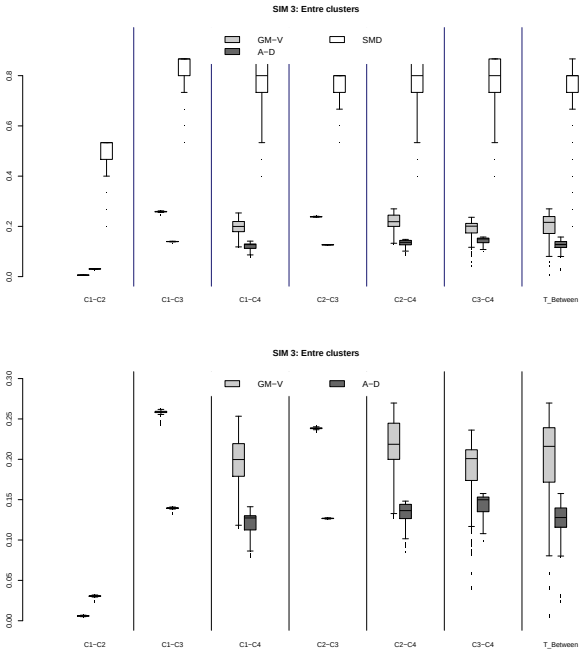

Figura A.6: Escenario 3: boxplot de distancias entre distintos conglomerados.

- **Escenario 4**

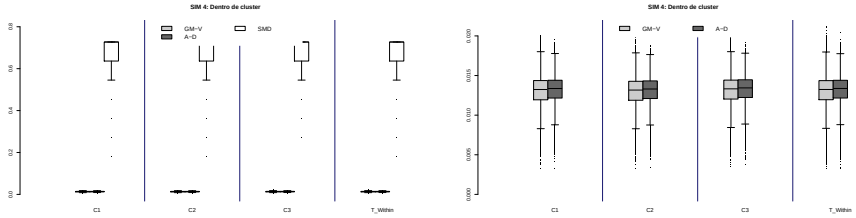

Figura A.7: Escenario 4: boxplot de distancias dentro de cada conglomerado.

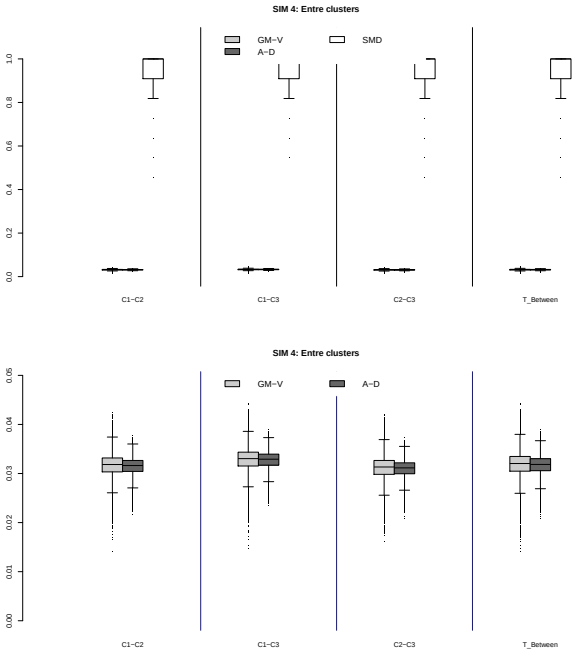

Figura A.8: Escenario 4: boxplot de distancias entre distintos conglomerados.

- **Escenario 5**

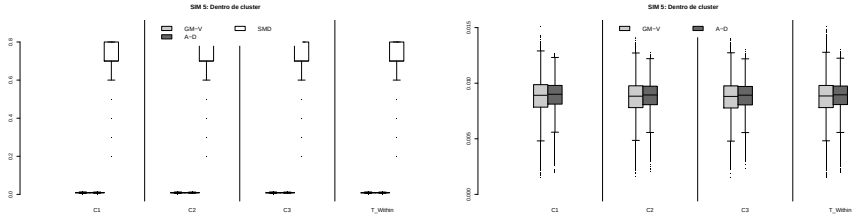

Figura A.9: Escenario 5: boxplot de distancias dentro de cada conglomerado.

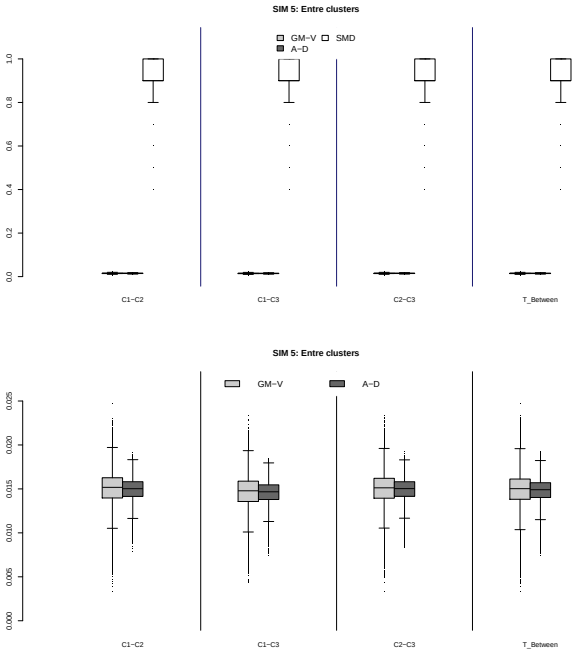

Figura A.10: Escenario 5: boxplot de distancias entre distintos conglomerados.

Referencias

Amir Ahmad and Lipika Dey. A k-mean clustering algorithm for mixed numeric and categorical data. *Data and Knowledge Engineering*, 63(2):503 – 527, 2007.

Paul S. Bradley and Usama M. Fayyad. Refining initial points for k-means clustering. In -, pages 91–99. Morgan kaufmann, 1998.

Igor Cadez, David Heckerman, Christopher Meek, Padhraic Smyth, and Steven White. Model-based clustering and visualization of navigation patterns on a web site. *Data Min. Knowl. Discov.*, 7(4): 399–424, 2003. ISSN 1384-5810.

Tak-chung Fu. A review on time series data mining. *Eng. Appl. Artif. Intell.*, 24(1):164–181, February 2011.

Manuel García-Magariños and José A. Vilar. A framework for dissimilarity-based partitioning clustering of categorical time series. *Data Mining and Knowledge Discovery*, pages 1–37, 2014. ISSN 1384-5810. doi: 10.1007/s10618-014-0357-y.

Martin Gavrilov, Dragomir Anguelov, Piotr Indyk, and Rajeev Motwani. Mining the stock market (extended abstract): Which measure is best? In *Proceedings of the sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD'00, pages 487–496, New York, USA, 2000. ACM.

Zhexue Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304, 1998.

T. Warren Liao. Clustering of time series data : A survey. *Pattern Recognition*, 38(11):1857–1874, 2005.

Christoph Pamminger. *bayesMCClust: Mixtures-of-Experts Markov Chain Clustering and Dirichlet Multinomial Clustering*, 2012. URL http://CRAN.R-project.org/package=bayesMCClust. R package version 1.0.

Christoph Pamminger and Sylvia Frühwirth-Schnatter. Model-based clustering of categorical time series. *Bayesian Anal.*, 5(2):345–368, 2010.

J.M. Pena, J.A. Lozano, and P. Larranaga. An empirical comparison of four initialization methods for the k-means algorithm, 1999.

H. Ralambondrainy. A conceptual version of the k-means algorithm. *Pattern Recogn. Lett.*, 16(11): 1147–1157, November 1995. ISSN 0167-8655. doi: 10.1016/0167-8655(95)00075-R. URL http://dx.doi.org/10.1016/0167-8655(95)00075-R.

William M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.