Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

ALGORITMO DE AGRUPACIÓN JERÁRQUICA BASADO EN AGREGACIÓN DE RANKINGS

David Martín Paule

Dirigido por
Noelia Rico Pachón
Pedro Huidobro Fernández

UNIVERSIDAD DE OVIEDO
Facultad de Ciencias
Grado en Matemáticas

15 de Febrero de 2022

Índice general

Capítulo 1

Introducción

Desde tiempos inmemorables, para la comprensión de nuestro entorno el ser humano utiliza la clasificación y la agrupación. La capacidad de agrupar todo tipo de entidades en función de su similitud es adquirida desde la infancia de forma natural. Imaginemos que damos a un niño pequeño una caja con diferentes piezas de madera de distintos tamaños y colores: círculos, estrellas, cuadrados, hexágonos; grandes y pequeños, de color amarillo, rojo y azul. Le dejamos solo media hora y le indicamos que, en la medida de lo posible, los agrupe. Aunque desconozca el nombre de los colores o de las formas, probablemente será capaz de agruparlas. Pero, ¿cómo las agruparía? puede hacerlo de maneras muy distintas y todas ellas válidas.

La tarea anterior es sencilla, pero para un humano. Transmitir nuestra manera de ver similitudes a un ordenador, para que este infiera agrupaciones, se convierte en una compleja labor. Si en vez de tener poliedros tuviésemos una serie de puntos, como en la Figura 1.1, intuitivamente nos sería muy fácil clasificarlos en tres grupos distintos. Ahora bien, a la hora de diseñar un programa que de manera autónoma sea capaz de ir agrupando esa serie de puntos, ¿cómo lo haríamos?, ¿qué criterios debemos mantener a lo largo de nuestro proceso para que tenga sentido nuestro agrupamiento?

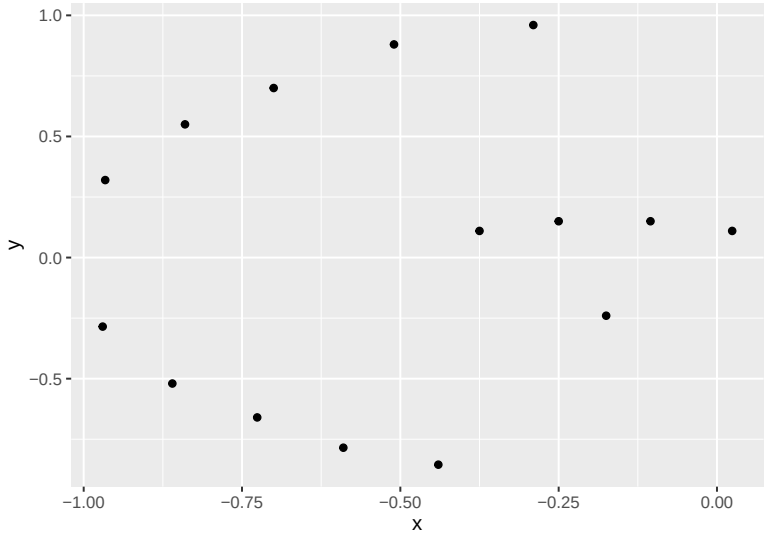

Figura 1.1: Ejemplo de puntos que pueden ser diferenciados en tres grupos diferentes de forma intuitiva.

De este ímpetu por encontrar grupos de elementos surge el análisis clúster [1], cuya finalidad dado un conjunto de datos es estudiar la creación de grupos de objetos del conjunto de datos que sean homogéneos. Es decir, se busca definir algoritmos que creen grupos de objetos del conjunto de datos, de tal forma que los objetos del mismo grupo sean lo más similares posibles y a la vez lo más diferentes posibles de los objetos pertenecientes a otros grupos. Existen diversos algoritmos de *clustering*, que pueden ser clasificados a grandes rasgos entre algoritmos de *particionado* o *agrupamiento*, dependiendo si la estrategia consiste en crear particiones iniciales o agrupar los objetos de forma iterativa, respectivamente. Otra clasificación de los algoritmos puede darse como *difuso (fuzzy)* o *duro (hard) clustering*, dependiendo de si en su definición un objeto puede pertenecer a un solo grupo o a más de uno. En este trabajo nos centraremos en el algoritmo de agrupamiento duro conocido como ***clustering* jerárquico**, que se trata de un algoritmo clásico de uso extensivo hoy en día debido a su simplicidad y flexibilidad computacional, lo cual permite su aplicación en problemas reales.

Este algoritmo se basa en la agrupación de objetos en función de su similitud. Consiste en agrupar los objetos del conjunto de datos de forma iterativa, buscando los clústers más similares y fusionándolos en un solo clúster, hasta obtener el número de grupos deseado. Para ser aplicado, requiere definir pre-

viamente la forma en la que se comparan los grupos de objetos con el fin de decidir los dos grupos que serán agrupados en solo uno. A lo largo de los años, se han ido desarrollando distintos *criterios de adhesión*, como por ejemplo el criterio *single* [2], el criterio *complete* [3] o el criterio *average* [4], entre otros. El algoritmo así como cada uno de estos criterios serán explicados con más detalle en el Capítulo 2.

Al trabajar con conjuntos de datos reales, lo común es encontrarse con que la mayoría de ellos tienen una estructura compleja, o simplemente desconocida. Por este motivo no es trivial decidir la mejor estrategia para la creación de grupos. Actualmente, una práctica común consiste en probar con cada criterio de adhesión, estudiando con detenimiento las propuestas de cada uno y finalmente escogiendo la partición adecuada de nuestro conjunto de datos según un criterio en concreto. Cada criterio tiene una manera distinta de atajar el problema de clusterización, y no existe un método que sea infalible para cualquier tipo de conjunto de datos.

Mi propuesta para este trabajo consiste en crear un nuevo método de *clustering* jerárquico a partir de los criterios de adhesión ya conocidos, diseñando un algoritmo que combine diferentes criterios con el fin de tomar la decisión de qué clústers deben ser fusionados de una forma más informada. La primera pregunta que surge es, ¿cómo podemos combinar los diferentes criterios de adhesión? La solución propuesta emplea una *agregación de rankings*, donde cada votante es cada uno de los criterios de adhesión a emplear, que ordenan los pares de clústers comparados en función de su similitud. El siguiente paso consiste en decidir cómo agregar la opinión en forma de ranking de cada criterio de adhesión con el fin de tomar una decisión consensuada.

El **recuento Borda** [5] es una técnica de agregación de votos basada en otorgar puntos a los candidatos en función de su posición en el ranking. Una variación de esta técnica conocida por la gran mayoría de personas es la utilizada en Eurovisión para decidir el ganador final en función a los puntos dados por cada país a cada uno de los países contrincantes. En el Capítulo 3 se define en profundidad esta ténica de agregación de rankings, que será utilizada en este trabajo para estudiar el nuevo método de clusterización propuesto que agregue la información de los criterios clásicos.

En el algoritmo propuesto, el recuento Borda se aplica sobre el conjunto de rankings individuales dados por cada criterio de adhesión de pares de

clústers más similares. Tras aplicar el recuento de Borda, se obtiene un único ranking consenso, resultado de agregar todos los rankings individuales y que, por lo tanto, tiene en cuenta los diferentes criterios. Este ranking se utilizará para tomar una decisión consensuada sobre los dos grupos de objetos que se juntan en uno solo, teniendo en cuenta las opiniones de cada uno de los métodos de adhesión. El método propuesto es detallado en el Capítulo 4.

Lo más interesante de esta propuesta de algoritmo de *clustering* jerárquico ascendente usando el recuento Borda es su aplicación es que lleva a unos resultados menos susceptibles al criterio de adhesión elegido, al agregar la opinión de diferentes criterios en cada paso.

El algoritmo introducido en este trabajo será además aplicado a un conjunto de datos real para ilustrar su comportamiento y ejemplificar la utilidad de la agregación de los diferentes criterios.

1.1. Objetivos y metodología

Este trabajo busca alcanzar los objetivos listados a continuación:

- Comprender el algoritmo de *clustering* jerárquico ascendente.
- Comprender los diferentes criterios de adhesión aplicables al *clustering* jerárquico ascendente.
- Comprender las técnicas de agregación de rankings y buscar su combinación consensuada.
- Diseñar un algoritmo que recoja las técnicas estudiadas: incorporación del recuento Borda al *clustering* jerárquico ascendente para la agregación de criterios de adhesión.
- Implementar el algoritmo de forma eficiente.
- Probar la fiabilidad y la utilidad de nuestro algoritmo, prestando atención en conseguir nuevos resultados.

A continuación, se detalla la metodología seguida para alcanzar los objetivos previamente establecidos:

1. Estudiar el algoritmo de *clustering* jerárquico ascendente.
2. Estudiar los principales criterios de adhesión utilizados en el algoritmo de *clustering* jerárquico ascendente.
3. Estudiar técnicas de agregación de órdenes.
4. Revisar la literatura sobre agregación de algoritmos de *clustering* ascendente.
5. Comparar diferentes propuestas de algoritmos para resolver el problema de agrupación ascendente de clústers incluyendo agregación de criterios, atendiendo a su eficiencia computacional.
6. Investigar la viabilidad de la implementación del algoritmo en los diferentes lenguajes de programación disponibles, fijándonos en las funcionalidades de cada uno.
7. Realizar el pseudocódigo del algoritmo elegido para simplificar su posterior desarrollo.
8. Implementar el algoritmo diseñado.
9. Seleccionar ejemplos clave que muestren la fiabilidad de nuestro método, así como los puntos débiles.
10. Refinar el algoritmo así como su implementación de manera progresiva para mejorar su eficiencia en la ejecución de problemas reales.
11. Aplicar el algoritmo a un conjunto de datos real.
12. Proponer soluciones para comportamientos observados en tiempo de ejecución como la resolución de empates.

1.2. Estructura del trabajo

Derivado de todo lo expuesto en este capítulo, el trabajo se estructura como sigue. En el Capítulo 2 se introducen conceptos básicos necesarios para comprender el agrupamiento jerárquico ascendente del análisis clúster, junto con algunos de los métodos que podemos emplear y cimentaremos nuestros conocimientos sobre estas técnicas mediante dos ejemplos. A continuación, en el Capítulo 3 se introduce el recuento de Borda. Después, en el Capítulo 4 veremos el grueso de nuestro estudio sobre el recuento Borda, y el diseño del algoritmo. En este capítulo también haremos hincapié en los diversos problemas que nos pueden surgir en su implementación algorítmica. Finalmente, en el Capítulo 5 realizamos una serie de comentarios conclusivos sobre la relevancia del trabajo desarrollado y las posibles líneas futuras de investigación.

1.3. Glosario

A modo de referencia para todo el documento, se proporciona en este primer capítulo el siguiente glosario, que introduce la terminología y notación empleadas.

Símbolo	**Significado**
c	Candidato o punto en el espacio.
$\mathcal{C}$	Conjunto de objetos para crear grupos.
$\mathscr{C}$	Conjunto de candidatos considerados en la clusterización.
$\mathscr{R}$	Perfil de rankings.
i	Reservado al número de iteración (subíndice).
j	Representa al criterio específico empleado para elaborar un ranking (superíndice).
n	Número de puntos en total en nuestra clusterización.
ν	Nivel de la clusterización.
L_ν	Número de parejas de clústers en el nivel ν.
M	Número de criterios específicos en total a considerar para elaborar un ranking.
k, l	Subíndices generales con significado según el contexto.
R^{c_i}	Rango del candidato c_i.
r_j	Ranking elaborado según el criterio j.
${}^{\nu}C_A$	Clúster A en el nivel ν de clusterización. Si no figura ningún nivel, nos referimos a un clúster en general.
${}^{\nu}(C_A, C_B)$	Pareja de clústers conformada por los clústers C_A y C_B en el nivel ν. Si no se menciona el nivel, es una generalización de cualquier pareja.
$p^j_{A,B}$	Puntuación en el recuento Borda de la pareja de clústers (C_A, C_B) para el ranking elaborado bajo el criterio j.

Tabla 1.1: Resumen de la notación utilizada a lo largo del documento.

Capítulo 2

Métodos de agrupamiento

En este capítulo se presentan algunos de los diferentes métodos de agrupación populares en la literatura, haciendo hincapié en las diferencias que hay entre ellos y presentando los conceptos teóricos fundamentales sobre los que se basa el método propuesto más adelante.

2.1. Introducción al *clustering* jerárquico

El **análisis clúster** o análisis de conglomerados es una técnica estadística cuya finalidad es inferir relaciones o estructuras entre los datos de un conjunto de datos $\mathcal{C}$. En este conjunto de datos los objetos de definen en base a una serie de variables. El resultado de este análisis es la clasificación de elementos se trata de unas *subpoblaciones*:

Definición 2.1. *Dado un conjunto $\mathcal{C}$ de datos sobre el que realizamos el análisis clúster, denominamos* ***subpoblación*** *a un subconjunto de elementos de $\mathcal{C}$, tales que estos tengan similitudes entre sí y diferencias con el resto de elementos del conjunto.*

De esta manera, el resultado del análisis será una serie de subpoblaciones que nos ayudarán a entender y agrupar la muestra de datos.

De entrada, no sabemos las características que conforman cada subpoblación, ya que es tarea del analista deducirlas. Para ello, recurriremos al uso de diferentes distancias y criterios para la formación de clústers.

Definición 2.2. *Un* ***clúster*** C_k *es un subconjunto homogéneo de puntos del conjunto de elementos de* $\mathcal{C}$*. Denominaremos por* $\mathscr{C}$ *al conjunto de clústers.*

Es necesario incidir en la diferencia entre clústers y subpoblaciones: está claro que toda subpoblación es un clúster, pero solo hablamos de subpoblaciones una vez finalizado nuestro análisis. El conjunto de datos puede ser dividido en diferente número de clústers, y la subpoblación coincide con la división óptima del conjunto. Por lo tanto, las subpoblaciones representan la clasificación final obtenida óptima para comprender la muestra, pues son clústers homogéneos en su contenido y distintos entre sí. La denominación de los clústers como subpoblaciones depende del criterio del analista, pues siempre se busca que sean explicativas de la muestra.

Para comenzar a agrupar los elementos del conjunto, vamos a recurrir al análisis jerarquizado, que es un método de análisis clúster iterativo en el cual, en cada iteración se agrupan dos clústers acorde a algún criterio prefedinido. Concretamente, en este trabajo nos centraremos exclusivamente en el análisis clúster jerarquizado ascendente.

Este tipo de análisis clúster consiste en partir de conjuntos unipuntuales, donde cada elemento del conjunto es considerado inicialmente un clúster de un único objeto. En cada iteración se irán fusionando, formando cada vez clústers de mayor tamaño. Nótese, que en un conjunto con n objetos, realizaremos $n-1$ iteraciones y comenzaremos con n clústers, uno por cada punto. Por cada iteración, se reduce el número de clústers en una unidad, quedando, después de la última, un único clúster final que englobará toda la muestra.

Como última definición introductoria, vamos a señalar la diferencia entre **niveles** y **iteraciones**. El nivel es una representación del número de clústers, partiendo siempre de n niveles y acabando con un único nivel como resultado de la agrupación iterativa. La primera iteración del algoritmo comienza en el nivel n, y cada iteración nueva se corresponde con el decremento en una unidad del nivel. A partir de ahora, al hablar de *nivel* en general, vamos a

representarlo con la letra ν, ya que así mantenemos la letra n estática para representar el número de puntos total del conjunto.

Ejemplo 2.1. *Vamos a tomar un ejemplo muy simple para ilustrar de forma intuitiva el comportamiento del algoritmo jerárquico y la forma de iterar por los distintos niveles. Tomemos el siguiente conjunto con animales $\mathcal{C}$:*

$$\mathcal{C} = \{Lince, Gato, Merluza, Tiburón\}$$

Imaginemos que queremos hacer un análisis clúster del conjunto $\mathcal{C}$ en función de lo similares que son los animales:
Partimos del nivel 4, pues al comienzo hay cuatro clústers, cada uno de ellos un conjunto unipuntual. En la primera iteración, notamos que el gato y el lince tienen mucho en común, pues ambos son felinos y tienen una apariencia física similar.
De esta manera, tras esta primera iteración pasamos al nivel tres, en el que tendremos los siguientes tres clústers:

$$C_{Felinos} = \{Gato, Lince\} \quad C_2 = \{Merluza\} \quad C_3 = \{Tiburón\}$$

Realizamos ahora la segunda iteración, donde relacionamos la merluza con el tiburón, al ser los dos peces (a pesar de la gran diferencia de tamaño). De esta manera, en el nivel 2, tendríamos dos clústers:

$$C_{Felinos} = \{Gato, Lince\} \quad C_{Peces} = \{Merluza, Tiburón\}$$

Tras esto, la tercera iteración resulta en el nivel 1, donde obtenemos un único clúster final, resultado de la unión de los dos clúster del nivel anterior:

$$C_{Animales} = C_{Felinos} \cup C_{Peces} = \mathcal{C}$$

Por último, si obtenemos como resultado final las dos subpoblaciones resultantes en el nivel 2, estas se corresponden con el clúster de felinos y de peces respectivamente. Esta clasificación en subpoblaciones responde a la idea intuitiva de agrupación de nuestro grupo de animales $\mathcal{C}$.

2.1.1. Similitud entre dos objetos

El algoritmo de *clustering* jerárquico requiere definir, en primer lugar, la forma de medir la similitud entre dos objetos del conjunto de datos.

A pesar de que en este trabajo nos centraremos exclusivamente en el estudio de los criterios de agregación de clústers (ver Sección 2.2), es importante señalar el impacto en el resultado final que puede surgir del empleo de diferentes similitudes [6].

Vamos a definir previamente el concepto de similitud: se trata de una manera de medir cómo de relacionados (o no) están dos conjuntos de datos entre sí. Su definición matemática es la siguiente [7]:

Definición 2.3. *Una función* $s\colon U \times U \to \mathbb{R}$ *se llama* ***similitud*** *(o disimilitud) cuando verifica las siguientes tres propiedades:*

1. $s(x, y) \leq s_0, \forall x, y \in U$
2. $s(x, x) = s_0$ *(con* $s_0 \in \mathbb{R}$ *un número arbitrario)*
3. $s(x, y) = s(y, x), \forall x, y \in U$

Un ejemplo de similitud podría ser un coeficiente de correlación. Una práctica común en el algoritmo de *clustering* jerárquico es utilizar la distancia entre los objetos del conjunto de datos, que son habitualmente definidos como vectores en $\mathbb{R}$, de tal forma que cuanto menor sea la distancia más similares se consideran los objetos. Cualquiera de las distancias mencionadas en la siguiente tabla puede ser encontrada en la literatura:

Distancia Euclídea	$d(\vec{x_i}, \vec{x_j}) = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2}$
Distancia de Manhattan	$d(\vec{x_i}, \vec{x_j}) = \sum_{k=1}^{p} \lvert x_{ik} - x_{jk} \rvert$
Distancia del supremo o de Chebyshov	$d(\vec{x_i}, \vec{x_j}) = \text{máx}_{k=1,\ldots,p} \lvert x_{ik} - x_{jk} \rvert$
Distancia de Mahalanobis	$d(\vec{x_i}, \vec{x_j}) = (\vec{x_i} - \vec{x_j}) S^{-1} (\vec{x_i} - \vec{x_j})$, siendo S la matriz de varianzas-covarianzas
Distancia de Bray-Curtis	$d(\vec{x_i}, \vec{x_j}) = \frac{\sum_{k=1}^{p} \lvert x_{ik} - x_{jk} \rvert}{\sum_{k=1}^{p} (x_{ik} + x_{jk})}$

En este trabajo, a la hora de ejemplificar el algoritmo propuesto nos centraremos exclusivamente en el uso de la distancia Euclídea como medida

de similitud entre objetos del conjunto de datos. El motivo de esta decisión es que la distancia Euclídea es, con mucha diferencia, la forma más empleada en la literatura para determinar los grupos de objetos utilizando el algoritmo de *clustering* jerárquico. Aún así, es importante destacar que el algoritmo propuesto puede ser aplicado con cualquier otra métrica de similitud entre objetos, como por ejemplo las propuestas en esta tabla.

La idea subyacente del análisis clúster es agrupar aquellos individuos con un alto grado de similitud entre ellos, separándolos de los individuos con los que se tenga un alto grado de diferencia.

Preparación de los datos

El uso de distancias como métrica de similitud requiere, para la gran mayoría de conjuntos, una preparación de los datos previa a la ejecución del algoritmo. Habitualmente las variables deberán ser normalizadas para garantizar que sean tratadas con el mismo grado de importancia por la distancia y no se presenten desequilibrios motivados por el uso de diferentes magnitudes entre variables. Además, si el conjunto de datos tiene una mezcla de variables cualitativas o cuantitativas, será necesaria su transformación, por ejemplo mediante el uso de variables dicotómicas.

Este paso es primordial, ya que si no lo realizásemos obtendríamos resultados confusos, en donde la interpretación de los datos se vería sesgada por la desproporción entre las variables.

2.2. Similitud entre clústers

El computo de la similitud entre dos objetos del conjunto de datos es trivial una vez establecida una de las métricas previamente definidas. A partir de la segunda iteración del algoritmo, donde ya existe al menos un clúster, es necesario determinar cómo la similitud entre objetos y clústers y cómo la similitud entre clústers y clústers es calculada. Varias preguntas deben

de ser resultas para establecer la forma de realizar este computo: ¿cómo se representa el clúster?, ¿se selecciona un único punto del clúster como representante?, ¿se consideran todos los puntos del clúster?

Los **criterios de adhesión** (conocidos con múltiples nombres de su traducción del inglés *linkage methods*) son diferentes técnicas cuya una única finalidad es determinar la similitud de los clústers. Hay una gran variedad de criterios a emplear. A grandes rasgos, podemos diferenciar los más populares en tres grupos:

- Los basados en comparar un par de objetos para cada par de clústers:
  - Criterio *single* (más similares de cada clúster).
  - Criterio *complete* (más diferentes de cada clúster).
- Los basados en seleccionar un único objeto como representante del clúster, y aplicar la similitud entre objetos para decidir la similitud entre clústers.
  - Criterio del centroide (*Unweighted pair group method with centroid clustering*, UPGMC).
  - Criterio de la mediana (*Weighted pair group method with centroid clustering*, WPGMC)
- Los que tienen en cuenta todos los objetos de cada clúster.
  - Criterio *average* (*Unweighted pair group method with arithmetic mean*, UPGMA).
  - Criterio de Ward.
  - Critero de McQuitty (*Weighted pair group method with arithmetic mean*, WPGMA).

En las siguientes subsecciones estudiaremos algunos de los criterios mencionados anteriormente, dando ejemplos y mostrando su funcionamiento.

Para facilitar la tarea, vamos a emplear siempre el mismo conjunto de puntos, que podemos ver representados gráficamente en la Figura 2.1 y cuyos datos se muestran definidos en la Tabla 2.1.

Puntos	Coordenadas	
	x	y
1	0.9	3.74
2	2	2
3	3.75	0.902
4	4	2.76
5	3	3.76
6	5.1	3.76

Tabla 2.1: Definición de conjunto de datos de ejemplo, representados de forma gráfica en la Figura 2.1.

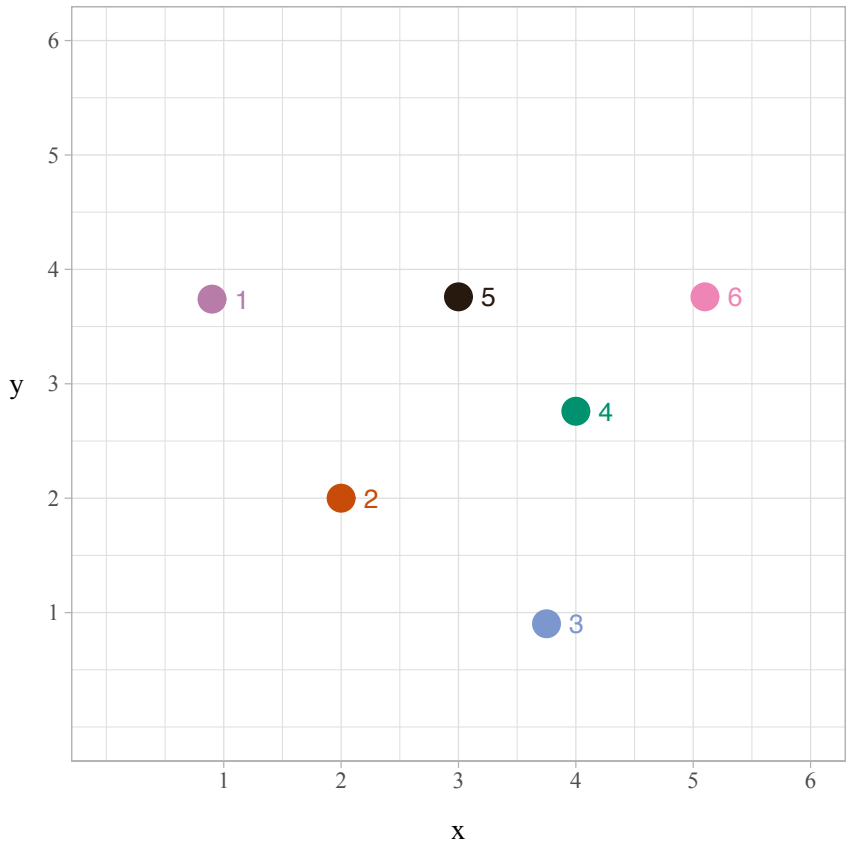

Figura 2.1: Representación gráfica del conjunto de datos presentado en la Tabla 2.1 que nos servirá de ejemplo para mostrar los diferentes métodos de agregación.

$C_2 \rightarrow$	2.058543				
$C_3 \rightarrow$	4.021763	2.065736			
$C_4 \rightarrow$	3.251215	2.139533	1.874365		
$C_5 \rightarrow$	2.100095	2.024253	2.954400	1.414214	
$C_6 \rightarrow$	4.200048	3.564772	3.160455	1.486607	2.100000
	$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$
	C_1	C_2	C_3	C_4	C_5

Tabla 2.2: Matriz de distancias para cada par de clústers definidos en el ejemplo representado en la Figura 2.1

.

Para este ejemplo, se utilizará la distancia Euclídea como similitud entre los objetos del conjunto de datos. Cuando dos objetos tienen menor distancia, se consideran más similares. En la Tabla 2.2 podemos encontrar las distancias entre los puntos. En esta tabla se utiliza la notación de clústers unipuntuales, para así estar en línea con la notación que seguiremos en el resto de trabajo. Por otra parte las diferentes distancias están sombreadas en una paleta de diferentes tonos de rojo, a mayor intensidad, menor distancia.

2.2.1. Estudio del criterio *single*

El criterio *single* o criterio de la distancia del salto mínimo, y define la distancia entre dos clústers cualesquiera como la distancia entre sus respectivos dos objetos más cercanos entre sí. Matemáticamente se expresaría así:

$$d(C_A, C_B) = \text{mín}\{d(a, b) \colon a \in C_A, b \in C_B\}$$

Este criterio funciona bien cuando las posibles subpoblaciones tienen formas irregulares o alargadas. Por el contrario, el uso de este criterio implica la posible fusión de clústers menos compactos, creando clústers donde dos objetos dentro de un mismo clúster pueden estar mucho más lejos entre sí que otros dos elementos de clústers distintos.

Observando la Figura 1.1, donde intuitivamente podemos deducir tres

subpoblaciones sencillamente:

1. Un primer clúster *norte*, con valores en el eje X comprendidos en $(-1, 0.25)$ y en el eje Y en $(0, 1)$.
2. Un segundo clúster *central*, que encontramos a la derecha de la representación gráfica, con valores en el eje Y contenidos entre $(-0.25, 0.25)$.
3. Un último clúster *sur*, contenido en el siguiente rectángulo: $(-1, -0.375) \times (-1, 0)$.

Notamos que los extremos izquierdos de los clústers *sur* y *norte* están más cerca entre sí que sus respectivos extremos derechos contenidos en los propios clústers.

Veamos un ejemplo sencillo de aplicación con las distancias de la Tabla 2.2 que provienen de la Figura 2.1.

Nota 2.2. *Tanto en este ejemplo como en el siguiente, como la finalidad es mostrar y ejemplificar el funcionamiento de los criterios, vamos a realizar algunas «concesiones»:*

- *A la hora de denotar los clústers, simplificaremos la notación respecto a la mostrada en el glosario y utilizada en otros puntos del documento. No indicaremos el nivel del clúster y por lo tanto se utilizará C_a en vez de ${}^{\nu}C_i$.*
- *Los clústers con múltiples puntos separados por comas en su subíndice (por ejemplo $C_{a,b,c}$) representan la unión en las iteraciones anteriores de los clústers contenedores de los respectivos puntos (de manera que $C_{a,b,c}$ indica que en las iteraciones anteriores se unieron C_a, C_b y C_c).*

A pesar de que esta notación es más sencilla, no podemos mantenerla durante el trabajo, pues la información sobre el nivel y los subíndices es de utilidad para futuras definiciones.

La primera iteración se crea un clúster a partir de los dos puntos más cercanos. De esta manera, uniremos C_4 y C_5, ya que son los que tienen la menos distancia (1.414214) y por lo tanto son los más similares. Es necesario apuntar que esta primera iteración es igual para todos los criterios, ya que los clústers solo contienen un objeto y por lo tanto no es necesario utilizar el criterio de adhesión con el fin de determinar cómo el clúster es representado.

El siguiente paso consiste en reinterpretar las distancias de la Tabla 2.2. Esto se basa en tomar las distancias desde los puntos 4 y 5 al resto y escoger las más pequeñas a cada uno de los otros clústers, de manera que realmente se esté considerado la distancia del objeto más cercano a cada uno de los otros clústers. A continuación se muestra la nueva tabla de distancias:

C_2	$\rightarrow$	2.058543			
C_3	$\rightarrow$	4.021763	2.065736		
$C_{4,5}$	$\rightarrow$	2.100095	2.024253	1.874365	
C_6	$\rightarrow$	4.200048	3.564772	3.160455	1.486607
		$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$
		C_1	C_2	C_3	$C_{4,5}$

Ahora, los clústers más similares son el recién creado $C_{4,5}$ con C_6 ya que tienen la menor distancia. Estos clústers se juntan en un único clúster y se repite el proceso detallado anteriormente, manteniendo la distancia mínima a cada uno de los otros clústers, para obtener la nueva matriz. La nueva tabla de distancias es:

C_2	$\rightarrow$	2.058543		
C_3	$\rightarrow$	4.021763	2.065736	
$C_{4,5,6}$	$\rightarrow$	2.100095	2.024253	1.874365
		$\uparrow$	$\uparrow$	$\uparrow$
		C_1	C_2	C_3

Ahora, en la siguiente iteración uniríamos el clúster C_3 con el cústers $C_{4,5,6}$. El resultado se uniría con el único clúster restante C_2 donde se alcanzaría un único clúster final que contiene todo el conjunto de datos $\mathcal{C}$.

La representación gráfica del resultado de la aplicación de los criterios se denomina **dendrograma**. Es un grafo con forma de árbol binario, donde en cada nivel se muestran los dos clústers que se han unido. De esta manera, tenemos tantas *raíces* del árbol como número de puntos desde el que partimos y el *tronco* del árbol representa el último clúster final que es el conjunto en su totalidad. El algoritmo de *clustering* jerárquico ascendente se aplica de abajo hacia arriba sobre el dendrograma. La altura de las raíces del dendrograma denota la distancia bajo la que se han unido, y es fácil deducir el orden.

Para ver el orden de estas uniones obtenidas con el criterio single, recurrimos al dendrograma, que está en la Figura 2.2.

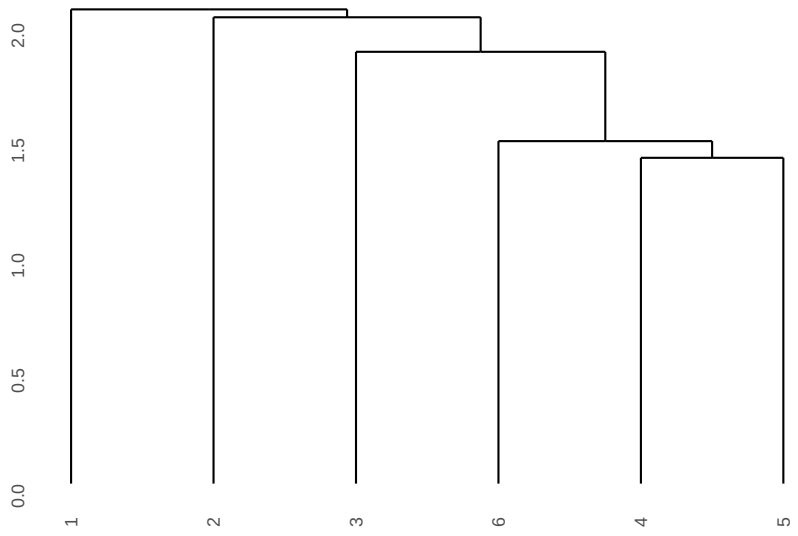

Figura 2.2: Dendrograma del análisis clúster de los puntos de la Tabla 2.1 para el criterio *single*.

2.2.2. Comparación con el criterio *complete*

El criterio *complete* propone la idea opuesta al criterio *single*. Esta vez, la similitud entre dos clústers se considera como la distancia entre sus dos punto más lejanos:

$$d(C_A, C_B) = \text{máx}\{d(a,b) \colon a \in C_A, b \in C_B\}$$

Así, este tipo de agregación es más *conservador* que el anterior, pues va a tender a grupos con diámetros similares, donde las distancias entre elementos sean homogéneas. Y esto también puede ser una desventaja, pues los valores atípicos pueden influenciar en el proceso de clusterización, generando anomalías.

Veamos cómo actuaría este criterio para los puntos de la Tabla 2.1.

La primera iteración, como ocurría con el criterio *single*, consiste en unir los puntos 4 y 5, resultándonos las siguientes distancias. Es importante incidir en que únicamente estamos reinterpretando las distancias de los puntos 4 y 5 al resto de puntos, escogiendo las mayores entre los dos puntos. Nótese el contraste con el previamente mostrado criterio *single*, donde se hacía lo opuesto. El resto de distancias se mantienen inalteradas:

C_2	$\rightarrow$	2.058543			
C_3	$\rightarrow$	4.021763	2.065736		
$C_{4,5}$	$\rightarrow$	3.251215	2.139533	2.954400	
C_6	$\rightarrow$	4.200048	3.564772	3.160455	2.100000
		$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$
		C_1	C_2	C_3	$C_{4,5}$

Ahora notamos que son los clústers C_1 y C_2 son los que tienen la menor distancia, uniéndolos en la segunda iteración. De nuevo recalcar la diferencia con el resultado obtenido en esta iteración del método *single*. Repetiríamos el proceso, llegando a las siguientes distancias:

C_3	$\rightarrow$	4.021763		
$C_{4,5}$	$\rightarrow$	3.251215	2.954400	
C_6	$\rightarrow$	4.200048	3.160455	2.100000
		$\uparrow$	$\uparrow$	$\uparrow$
		$C_{1,2}$	C_3	$C_{4,5}$

Análogamente, seguiríamos, ahora uniendo los clústers $C_{4,5}$ y C_6. A continuación, en la Figura 2.3 tenemos el respectivo dendrograma obtenido con

el criterio de adhesión *complete*. Si nos fijamos en el eje izquierdo, veremos que los valores son superiores que en el dendrograma del criterio *single* (ver Figura 2.2). Esto se debe a que, al escoger el criterio *complete* la distancia máxima, tendremos raíces del dendrograma más largas. También podemos observar este fenómeno cromáticamente, pues a medida que vamos reinterpretando las distancias en las sucesivas iteraciones, podemos apreciar como en el nivel 3 del criterio *single* encontramos, en general, rojos más intensos que en el respectivo nivel del criterio *complete*.

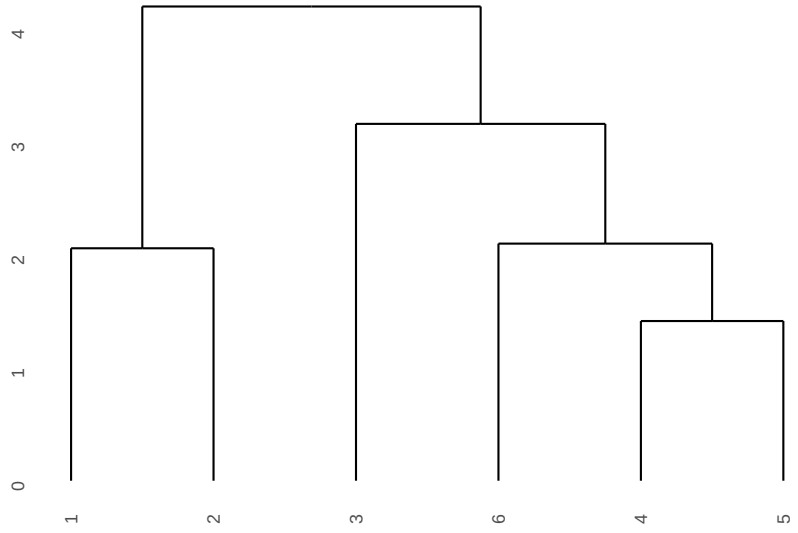

Figura 2.3: Dendrograma del análisis clúster de los puntos de la Tabla 2.1 para el criterio *complete*.

2.2.3. Ejemplo de aplicación con criterios basados en distancias promedio

Vamos ahora a estudiar tres subgrupos de criterios:

Criterios *average* y de McQuitty

Para estos dos criterios, la distancia entre clústers es siempre la media de las distancias entre los elementos de cada clúster con respecto al otro. La diferencia entre ambos reside en que para criterio *average* se tiene en cuenta el cardinal (o tamaño) de los clústers, mientras que para el criterio de McQuitty se realiza una media sin que importe el tamaño.

- Criterio *average*: $d(C_A, C_B) = \frac{1}{|C_A| \cdot |C_B|} \sum_{a \in C_A} \sum_{b \in C_B} d(a, b)$
- Criterio de McQuitty [1]: $d(C_A \cup C_B, C_D) = \frac{d(C_A, C_B) + d(C_B, C_D)}{2}$

Al recurrir a este tipo de fórmulas, vamos a necesitar en cada iteración un gasto computacional elevado necesario para realizar las medias. Veamos su representación en dendrogramas.

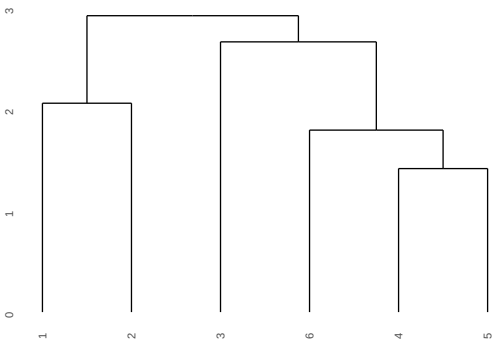

Figura 2.4: Dendrograma para el criterio *average*.

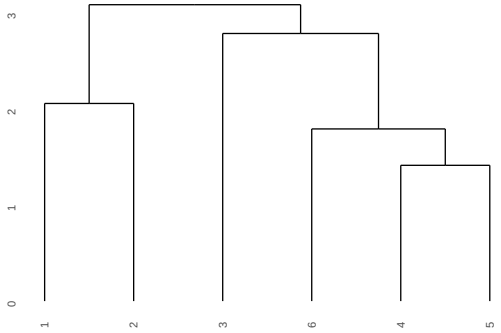

Figura 2.5: Dendrograma para el criterio de McQuitty.

Estos criterios suelen responder muy bien cuando tratamos de encontrar grupos aproximadamente esféricos.

[1] En el criterio de McQuitty, hay que basarse en las distancias calculadas en la iteración anterior. De esta manera hay que realizar modificaciones constantes a la matriz de distancias. Por otra parte, da resultados distintos al criterio *average* ya que la media no es asociativa: dados a, b y c no siempre se tiene que $\frac{\frac{a+b}{2}+c}{2} = \frac{\frac{a+c}{2}+b}{2} = \frac{\frac{b+c}{2}+a}{2}$.

Criterios del centroide y la mediana

Estos dos métodos no estudian los clústers en sí, sino su centroide (o centro de masas) o mediana geométrica. La distancia entre dos clústers es simplemente la distancia entre sus centroides o medianas, reduciendo cada clúster a un único punto simbólico.

En cuanto al criterio del centroide, al tratarse del centro de masas, el tamaño del clúster juega un rol importante a la hora de considerar el punto. Sin embargo, esto no ocurre en el criterio de la mediana, que se basa en un concepto puramente geométrico donde no interfieren los cardinales de cada clúster.

Es importante mencionar que estos criterios no son monotónicos: si nos fijamos en los respectivos dendrogramas (ver Figuras 2.6 y 2.7), notaremos como difieren del resto al encontrarse una especie de intersecciones entre las raíces. Esto se debe a una **inversión** del dendrograma, y es un fenómeno que puede ocurrir bajo estos dos métodos.

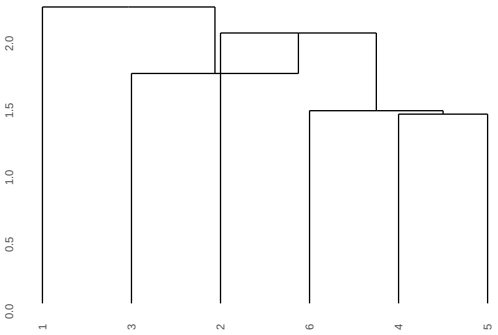

Figura 2.6: Dendrograma para el criterio del centroide.

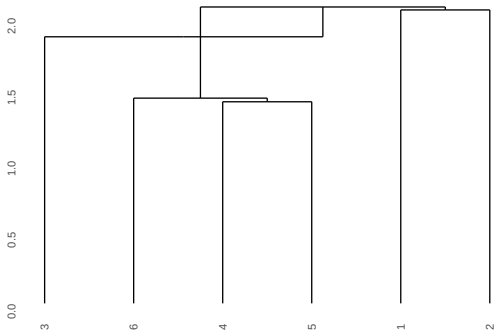

Figura 2.7: Dendrograma para el criterio de la mediana.

Si por ejemplo tomamos exclusivamente los puntos 1, 2 y 5 de la Figura 2.1, tenemos que representan un triángulo casi isósceles. En la primera iteración uniríamos el punto 2 con el 5 (al ser los más próximos), y obtendríamos su centroide/mediana en el punto medio del segmento que los une, que vamos a llamar M_1. Ahora bien, la distancia de M_1 a 1 es inferior que la de 2 a 5 y 1 a 5. De esta manera, obtenemos una nueva distancia mínima aún menor que la distancia mínima de la iteración anterior, teniendo que hacer

una raíz del dendrograma menos alta que la raíz de partida.

Criterio de Ward

Este criterio está basado en el análisis ANOVA, pues une aquellos clústers cuya fusión produzca el menor incremento en el valor total de la suma de los cuadrados respecto al centro del clúster resultante. Su expresión matemática es la siguiente[2][8]:

$$D(C_A, C_B) = \text{ECM}[C_A \cup C_B] - (\text{ECM}[C_A] + \text{ECM}[C_B])$$

Como vemos, este criterio requiere una carga computacional bastante elevada, pero cuando se trata de buscar grupos esféricos es el más empleado.

Es importante señalar que este criterio, junto con el del centroide y la mediana, deberían reservarse cuando empleamos la distancia Euclídea. El uso de otro tipo de distancia podría sesgar nuestros resultados.

A continuación, en la Figura 2.8 encontramos su respectivo dendrograma.

[2] ECM son las siglas que representan el *error cuadrático medio*, que el criterio de Ward trata de minimizar. Su fórmula de cálculo [9], adaptada para un clúster cualquiera $C_A = \{c_1, \ldots, c_n\}$, es la siguiente:

$$\text{ECM}[C_A] = \sum_{i=1}^{n} c_i^2 - \frac{1}{n}\left(\sum_{i=1}^{n} c_i\right)^2$$

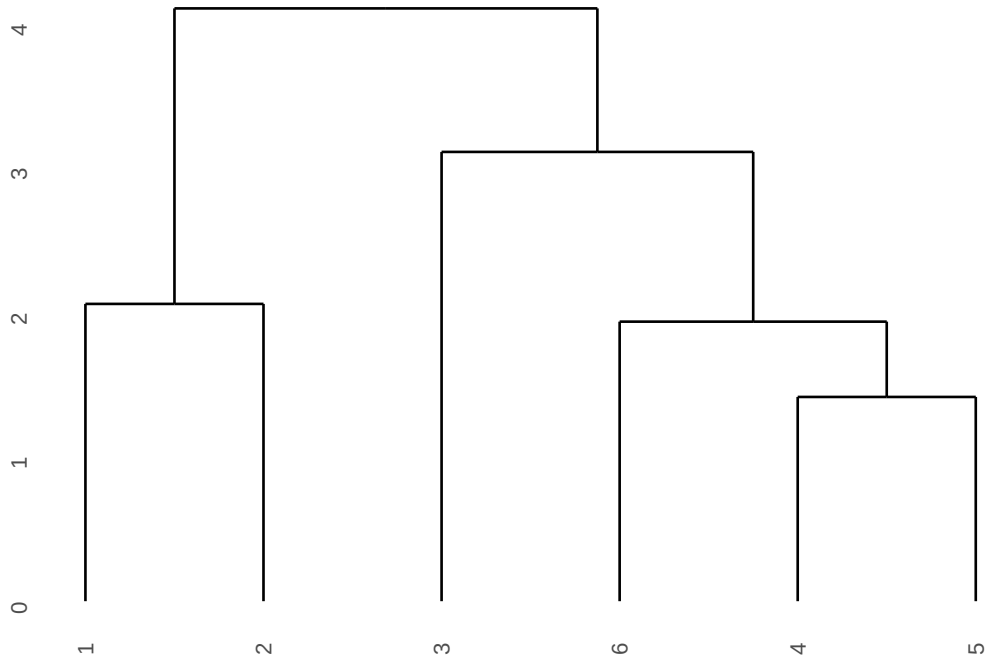

Figura 2.8: Dendrograma del análisis clúster de los puntos de la Tabla 2.1 para el criterio de Ward.

Capítulo 3

Recuento Borda en la agregación de rankings

El objetivo de este capítulo es introducir los conceptos necesarios para abordar la agregación de diferentes opiniones dadas por varios votantes sobre un conjunto de candidatos. Denominaremos **candidatos** a los elementos de un conjunto sobre el que se quiere determinar un ganador.

Definición 3.1. *Dado un conjunto $\mathscr{C} = \{c_1, \ldots, c_n\}$ de candidatos, el **rango** R^{c_l} de un candidato c_l, $l \in \{1, \ldots, n\}$, dentro del conjunto $\mathscr{C}$ es el número ordinal que representa la posición de ese candidato cuando ordenamos el conjunto en base a un criterio específico.*

El concepto de rango es meramente introductorio para nuestra siguiente definición, sobre la que se basa este trabajo:

Definición 3.2. *Un **ranking** r_j es la representación ordenada de los candidatos del conjunto $\mathscr{C}$ en función de su rango; donde $c_k \succ c_l$ representa que el rango del candidato c_k es menor que el de c_l, y $c_k \sim c_l$ que sus rangos son iguales, para todo $c_k, c_l \in \mathscr{C}$ y $k, l \in \{1, \ldots, n\}$, con $k \neq l$.*

Debe notarse que cada permutación π del conjunto $\mathscr{C}$ de candidatos puede representarse en un ranking. De esta manera, a partir de un conjunto $\mathscr{C}$

cualquiera de n candidatos, el número de rankings posibles a representar será $\text{card}(S_n) = n!$, siendo S_n el grupo simétrico cuyos elementos son todas las permutaciones posibles de los n candidatos.

A continuación se proporciona un ejemplo para comprender el concepto de ranking:

Ejemplo 3.1. *Sean $\mathscr{C} = \{$Sevilla, Tenerife, Chipre, Cancún, Marrakech$\}$ las diferentes opciones de viaje que barajan el grupo de amigos de Patricia para ir de vacaciones. Ella prefiere destinos de playa, y después de eso, cuanto más cerca de Oviedo mejor. De esta manera, el ranking del conjunto $\mathscr{C}$ según el criterio de Patricia sería:*

$$r_{Patricia} \equiv Tenerife \succ Chipre \succ Cancún \succ Sevilla \succ Marrakech$$

Análogamente, a partir de este ranking podemos obtener los rangos de cada destino:

$$R^{Patricia} = \{R^{Sevilla} = 4, R^{Tenerife} = 1, R^{Chipre} = 2, R^{Cancún} = 3, R^{Marrakech} = 5\}$$

Trabajaremos con diferentes rankings, relacionados con las distancias entre clústers según los diferentes criterios de agregación. No obstante, ese no es el único uso que le daremos, pudiendo elaborar rankings en distintas situaciones a partir de diferentes criterios.

Definición 3.3. *Denominaremos **votante** j al criterio específico bajo el cual se ha elaborado un ranking r_j.*

Habitualmente, varios votantes expresarán rankings sobre el conjunto de candidatos. Cada votante tendrá su propio criterio, que será reflejado en su ranking y puede coincidir o no con el de otros votantes.

Definición 3.4. *Llamamos **perfil de rankings** $\mathscr{R}$ a un conjunto de rankings sobre el conjunto de candidatos $\mathscr{C}$ que cumplan la siguiente condición: cada ranking r_j del perfil es una representación ordenada de todos los candidatos del conjunto $\mathscr{C}$ donde se establece un orden para cada par de candidatos.*

Para ilustrar el concepto de perfil de rankings se muestra el siguiente ejemplo, que proporciona nuevas clasificaciones de los diferentes candidatos del ejemplo anterior, posibilitando la creación de nuestro primer perfil de rankings.

Ejemplo 3.2. *Si creásemos un perfil de rankings $\mathscr{R}$ del conjunto $\mathscr{C}$ del ejemplo anterior, incluyendo como nuevos votantes al resto del grupo de amigos, que tienen las siguientes preferencias:*

- *Borja: no quiere salir de la Unión Europea, y si puede ser, ni salir de España ni coger un avión.*
- *Carla: prefiere ir a un sitio donde no hablen castellano, y le llama mucho la atención Marruecos.*
- *Hugo: Quiere irse fuera de España, cuanto más lejos mejor, priorizando destinos de playa.*

De esta manera, podemos construir sus rankings:

$$r_{Borja} \equiv Sevilla \succ Tenerife \succ Chipre \succ Cancún \sim Marrakech$$
$$r_{Carla} \equiv Marrakech \succ Chipre \succ Cancún \sim Sevilla \sim Tenerife$$
$$r_{Hugo} \equiv Cancún \succ Chipre \succ Marrakech \succ Sevilla \sim Tenerife$$

Así, nuestro perfil de rankings sería $\mathscr{R} = \{r_{Patricia}, r_{Borja}, r_{Carla}, r_{Hugo}\}$

De nuestro ejemplo notamos que no se puede hacer prevalecer el criterio de un votante sobre el resto. Para llegar a un candidato final, debemos considerar todos los rankings del perfil. Se trata de llegar a un consenso teniendo en cuenta todas las opciones de cada votante. De aquí surge la siguiente definición.

Definición 3.5. *Un **método de agregación de rankings** es un procedimiento matemático para determinar un único ranking a partir de un perfil de ranking $\mathscr{R}$. Este ranking debe de resumir la opinión de todos los votantes para obtener un ranking ganador que establezca un consenso de los diferentes criterios.*

A lo largo de la historia han sido propuestos numerosos métodos de agregación de rankings, estudiados en el campo de teoría de elección social. A lo largo de este trabajo nos centraremos en el método conocido como *recuento de Borda*. Este método es presentado en la siguiente sección.

3.1. Terminología empleada en el método de recuento Borda

El *recuento de Borda* se trata de un método muy intuitivo, basado en obtener un único ranking del perfil de rankings por un sistema individual de puntos, donde cada candidato recibe una puntuación en cada uno de los rankings en base a su posición. Por su simplicidad y eficiencia computacional, el recuento de Borda es uno de los más utilizados y se puede encontrar en elecciones como por ejemplo la de las micronaciones del Pacífico como Nauru o Kiribati, o una adaptación de este, en la competición paneuropea *Eurovisión*.

Definición 3.6. *El* ***recuento Borda*** *es un método de agregación de rankings: a cada candidato en cada ranking se le asigna una puntuación en base a su rango, con una única condición: a menor rango, mayor puntuación. A partir de las puntuaciones totales de cada candidato se obtiene el nuevo ranking agregado.*

El siguiente ejemplo utiliza el perfil de rankings obtenido en los ejemplos anteriores y aplica sobre él el método de recuento Borda. Se muestra cómo asignar puntuaciones en base a los rankings de cada votante.

Ejemplo 3.3. *Vamos a realizar el recuento Borda a nuestro perfil de rankings $\mathscr{R}$ del ejemplo anterior. Para hacer más comprensiva nuestra tarea vamos a representarlo en una tabla. Además, vamos a recurrir a la versión clásica del recuento Borda, asignando, si no existen candidatos con el mismo rango: una puntuación de 0 al candidato final del ranking, 1 punto al anterior, 2 puntos al antepenúltimo, ..., así hasta llegar al primer elemento del ranking que tendrá una puntuación de $n-1$, siendo n el número el número de candidatos total.*

En el hipotético caso de que haya candidatos del mismo rango, les asignaríamos una puntuación igual a la media de las puntuaciones máxima y mínima no otorgadas.

Ranking	*Tenerife*	*Sevilla*	*Marrakech*	*Cancún*	*Chipre*
$r_{Patricia}$	*4*	*1*	*0*	*2*	*3*
r_{Borja}	*3*	*4*	*0.5*	*0.5*	*2*
r_{Carla}	*1*	*1*	*4*	*1*	*3*
r_{Hugo}	*0.5*	*0.5*	*2*	*4*	*3*
Total	*8.5*	*6.5*	*6.5*	*7.5*	*11*
Ranking agregado	$Chipre \succ Tenerife \succ Cancún \succ Sevilla \sim Marrakech$				

Como podemos observar, en el ranking final agregado nos encontramos con que la opción ganadora es Chipre. Sin embargo, no fue la candidatura principal de ninguno de los votantes: aquí vemos el potencial de este tipo de recuento, donde prima la búsqueda del ***consenso*** *entre los votantes.*

3.2. Algoritmo para el recuento Borda

En esta sección vamos a ejemplificar cómo programar el recuento Borda de una forma eficiente. Más adelante, en el siguiente capítulo, se detallará la aplicación del recuento Borda en el algoritmo de agrupamiento de clústers. El recuento de Borda es adecuado para esta tarea debido a su eficiencia computacional [10], ya que es posible calcular el ranking de Borda de un perfil de rankings en tiempo polinomial, a diferencia de lo que ocurre con otras reglas de agregación.

1. Tomamos el perfil de rankings y consideramos el número n de candidatos en $\mathscr{C}$.
2. Para cada ranking r_j contenido en el perfil de rankings $\mathscr{R}_i$, realizaremos:
   - A cada candidato del ranking se le asigna una puntuación: al último elemento 0 puntos, al penúltimo 1, antepenúltimo 2 y así sucesivamente hasta llegar al comienzo del ranking, donde al segundo elemento le concederemos una puntuación de $L_\nu - 2$ puntos y, finalmente, al primer elemento $L_\nu - 1$ puntos.
     En caso de empates, la puntuación asignada será la media entre la puntuaciones máxima y mínimas que no nos es posible otorgar.

3. Se suman las puntuaciones de cada candidato en cada ranking para obtener una puntaución final de cada candidato.

4. Se hace un ranking final ordenando los candidatos en base a su puntuación final de mayor a menos.

5. El candidato ganador es el candidato en la primera posición del ranking obtenido, es decir, el candidato con más puntos.

3.2.1. Optimización computacional del método de Borda

El algoritmo para calcular el ranking de Borda puede ser simplificado utilizando una versión simplificada que resume la información del perfil de rankings en forma de matriz.

Esta matriz contiene la comparación por pares de cada par de candidatos en $\mathscr{C}$. Se trata de una matriz de dimensiones $n \times n$. Para cada par de candidatos, $c_i, c_j \in \mathscr{C}$, la matriz contiene en el elemento $m_{i,j}$ de la i-ésima fila y j-ésima columna el número de veces que el candidato c_i aparece en una posición mejor en el perfil que el candidato c_j. Es decir, para cada ranking, si $c_i \succ c_j$ se añadirá un punto, si $c_i \succ c_j$ se añadirán 0.5 puntos y si $c_j \succ c_i$ se añadirán 0 puntos en la posición $m_{i,j}$.

Con esta representación por pares, la puntuación de cada uno de los candidatos puede ser calculada teniendo en cuenta la suma de las filas correspondientes a cada candidato, ya que en cada fila se encuentra el número de veces que el candidato es preferido sobre el resto de candidatos.

Por lo tanto, disponer de los perfiles de rankings en este formato simplificado, reduce la cantidad de tiempo necesaria para calcular el ranking de Borda.

Ejemplo 3.4. *Continuando con los ejemplos anteriores, para el ranking dado por Patricia obtendríamos la siguiente comparación por pares:*

La primera posición es ocupada por Tenerife, por lo que Tenerife es con-

	Tenerife	Sevilla	Marrakech	Cancún	Chipre
Tenerife	0	1	1	1	1
Sevilla	0	0	1	0	0
Marrakech	0	0	0	0	0
Cancún	0	1	1	0	0
Chipre	0	1	1	1	0

siderada mejor que todos los demás y por eso tiene un punto frente al resto de alternativas, A continuación aparece Chipre, que vence a todas menos tenerife. Cancún es mejor que Sevilla y Marrakech y por lo tanto Sevilla solo es mejor que Marrakech. Marrakech está en la última posición del ranking y por lo tanto su correspondiente fila es todo 0. Haciendo esto con todos los perfiles obtenemos la siguiente matriz:

	Tenerife	Sevilla	Marrakech	Cancún	Chipre	Total
Tenerife	0	2	2	2.5	2	8.5
Sevilla	2	0	2	1.5	1	6.5
Marrakech	2	2	0	1.5	1	6.5
Cancún	1.5	2.5	2.5	0	1	7.5
Chipre	2	3	3	3	0	11

Donde de nuevo obtenemos que:

$$Chipre \succ Tenerife \succ Cancún \succ Sevilla \sim Marrakech$$

.

Capítulo 4

Introducción del recuento Borda en el algoritmo de *clustering* jerárquico ascendente

A lo largo de este capítulo se presenta el algoritmo propuesto en el trabajo y se estudian a fondo los posibles problemas que puedan ocurrir durante su ejecución. Además, se presentan dos ejemplos para mostrar su utilidad, ilustrando como los resultados obtenidos varían respecto a los obtenidos con los métodos de adhesión clásicos.

4.1. Propuesta de algoritmo

En esta sección se presenta el algoritmo para introducir el recuento Borda en el método de *clustering* jerárquico ascendente con el fin de agregar la información proporcionada por los diferentes criterios de adhesión. A partir de ahora, es necesario contextualizar el problema, identificando quién serán los candidatos y votantes del recuento Borda. Las parejas de clústers forman los candidatos, ya que son las consideradas para unirse en cada iteración entra las que hay que determinar el ganador que se unirá finalmente. Cada criterio de adhesión será un votante, que genera un rankings ordenando los

candidatos (i.e. pares de clústers) de menor a mayor proximidad, según la distancia empleada, o lo que es lo mismo, de tal forma que los pares con mayor similitud están en posiciones mejores que los pares con peor similitud.

1. En primer lugar, leemos los datos de entrada y aplicamos la distancia Euclídea, con la que obtenemos la matriz D de distancias entre de distancias entre los objetos del conjunto de datos.

2. Se definen los métodos a emplear μ. Además, el número de objetos se representa con la letra n.

3. Realizamos la primera iteración: buscamos en la matriz triangular superior de D el valor más pequeño[1], que representará la pareja de clústers más cercana.

4. Unimos la pareja de clústers más cercana y almacenamos la información en la salida.

5. Iniciamos ahora un bucle, tal que para cada iteración restante $i \in \{2, 3, \ldots, n-2\}$:

   a) Se crea un perfil de rankings vacío $\mathscr{R}_i$.

   b) Se hace, para cada método a emplear ($\forall$*método* $\in \mu$ indicado en la entrada):

      1) Se transforma o reinterpreta la matriz de distancias D según el método:

         - Para métodos que no requieren recálculo de distancias

           a' Se reinterpretan las distancias de la matriz, manteniendo para cada par de clústers solo los elementos de la matriz relevantes acorde con el criterio utilizado.

         - Para los métodos que identifican el clúster mediante un único punto

           a' Se calculan los nuevos objetos que representan al clúster y con ellos se computa la nueva matriz de distancias.

[1] El hecho de fijarnos únicamente en la matriz triangular superior se debe a dos razones: la matriz de distancias es una matriz simétrica, en donde la diagonal principal contiene únicamente 0. Por otra parte, al escoger la matriz triangular superior, tenemos que cualquier elemento $a_{k,l}$ por encima de la diagonal principal va a tener un valor de k inferior al de l, facilitando así que en las parejas de clústers la pareja a la izquierda tenga un índice más bajo que la de la derecha, asegurando la integridad del proceso.

2) Se ordena de menor a mayor todos los elementos a considerar que estén contenidos en la matriz triangular superior de D, y se obtiene su posición dentro de la matriz. Esta serie de posiciones ordenadas representa en realidad las propuestas de parejas de clústers a unir del método, y se introducirá en el perfil de rankings.

c) Se realiza el recuento Borda del perfil de rankings obtenido.

d) Se selecciona el par ganador del ranking obtenido con el recuento de Borda y se fusionan los dos clústers recogidos en el par en un único clúster.

6. Finalmente, la última iteración es trivial y representa la unión de los únicos dos clústers resultantes.

7. Se devuelve la salida del método indicando en cada iteración la pareja de clústers a unir.

El algoritmo propuesto en este trabajo puede ser usado con cualquier similitud de las previamente definidas, así como con cualquier método de adhesión. La traducción de pseudocódigo a implementación no es trivial, y requiere tener en cuenta muchos aspectos de eficiencia y de gestión de memoria. Tras un estudio en términos computacionales de los criterios de adhesión introducidos en la Sección 2.2, se ha decidido implementar el algoritmo propuesto para los criterios *single*, *complete* y *average* ya que no requieren ninguna modificación de la matriz de comparación por pares entre los objetos originales del conjunto de datos. En los ejemplos y aplicaciones dadas en este trabajo se ilustrará el funcionamiento del algoritmo utilizando la agregación con el recuento Borda de dos o tres de estos criterios.

4.1.1. Abreviación del recuento Borda

El hecho de realizar un recuento Borda en todas las iteraciones implica un gasto en tiempo de ejecución. En algunas iteraciones, este recuento puede evitarse.

En el caso de que todos los rankings den el mismo par de clústers en primera posición, el ganador del recuento de Borda será este par y no es

necesario pasar por el proceso de otorgar puntos al resto de candidatos. Esto puede utilizarse como una precondición en la implementación del algoritmo con el fin de ahorrar tiempo de ejecución.

4.2. Aplicación del algoritmo propuesto

Vamos a particularizar lo visto en la Sección 3.2. Recalcar, como se había comentado previamente, que en este contexto los candidatos considerados por el recuento Borda serán, en cada iteración, las posibles parejas de clústers. Por otra parte, cada votante en realidad es un criterio de adhesión, y el ranking representa dado por cada criterio ordena las parejas de clústers candidatas, de más similares (menor distancia) a menos similares (mayor distancia). Como la forma de medida de la similitud entre clústers varía dependiendo del criterio de adhesión, se busca agregar la información recogida en los distintos rankings.

4.2.1. Ejemplo simple

Vamos a realizar un ejemplo simple, con seis objetos definidos en la Tabla 4.1 y representados en la Figura 4.1. Se ejemplificará la aplicación del algoritmo propuesto agregando los criterios *single* y *complete* empleando el recuento de Borda.

Puntos	Coordenadas	
	x	y
1	4	2
2	3	5
3	5	2
4	8	7
5	8	2
6	6	6

Tabla 4.1: Definición de conjunto de datos de ejemplo, representados en la Figura 4.1.

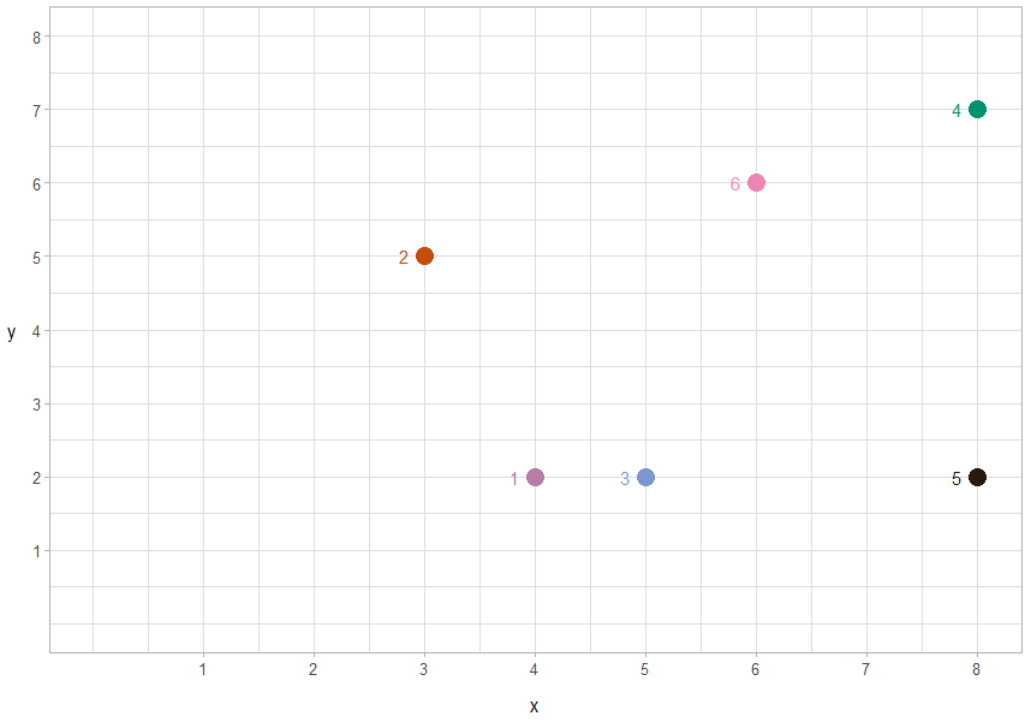

Figura 4.1: La muestra de objetos que nos servirá de ejemplo para mostrar los diferentes métodos de agregación.

En la Tabla 4.2 de la página siguiente podemos ver las distintas distancias entre elementos, vislumbrando cual será nuestra primera pareja de clústers a unir.

Empecemos la agrupación, paso a paso, mostrando detalladamente el ranking obtenido en cada nivel.

Nivel 6 En la primera iteración la tarea es muy sencilla. Únicamente debemos encontrar aquellos dos elementos más cercanos. De entrada, podemos hacer el primer ranking general de los clústers, que nos servirá para más adelante:

$$^6(C_1, C_3) \succ {}^6(C_4, C_6) \succ {}^6(C_3, C_5) \succ {}^6(C_1, C_2) \sim {}^6(C_2, C_6) \succ$$
$$^6(C_2, C_3) \succ {}^6(C_1, C_5) \succ {}^6(C_3, C_6) \succ {}^6(C_1, C_6) \sim {}^6(C_5, C_6) \succ$$
$$^6(C_4, C_5) \succ {}^6(C_2, C_4) \succ {}^6(C_3, C_4) \sim {}^6(C_2, C_5) \succ {}^6(C_1, C_4)$$

Los clústers a unir son los recogidos en el par ganador: $^6(C_1, C_3)$. Todos

los criterios dan este mismo ranking, ya que hasta el momento hay exclusivamente clústers unipuntuales. Es a partir de este punto en el que entrarán en juego los diferentes criterios de adhesión. Por ese motivo en este primer nivel puede omitirse la agregación de rankings, ya que llevaría al mismo resultado obtenido que con un solo ranking.

$C_2 \rightarrow$	3.162278				
$C_3 \rightarrow$	1.000000	3.605551			
$C_4 \rightarrow$	6.403124	5.385165	5.830952		
$C_5 \rightarrow$	4.000000	5.830952	3.000000	5.000000	
$C_6 \rightarrow$	4.472136	3.162278	4.123106	2.236068	4.472136
	$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$
	C_1	C_2	C_3	C_4	C_5

Tabla 4.2: Matriz de distancias para cada par de clústers definidos en el ejemplo representado en la Figura 4.1

.

Nivel 5 Renombramos de la siguiente manera nuestros clústers:

${}^6C_1 \rightarrow {}^5C_1$	${}^6C_4 \rightarrow {}^5C_3$
${}^6C_2 \rightarrow {}^5C_2$	${}^6C_5 \rightarrow {}^5C_4$
${}^6C_3 \rightarrow {}^5C_1$	${}^6C_6 \rightarrow {}^5C_5$

- Método *single*:

$$ {}^5(C_3, C_5) \succ {}^5(C_1, C_4) \succ {}^5(C_1, C_2) \sim {}^5(C_2, C_5) \succ {}^5(C_1, C_5) \succ {}^5(C_4, C_5) \succ {}^5(C_3, C_4) \succ {}^5(C_2, C_3) \succ {}^5(C_1, C_3) \sim {}^5(C_2, C_4) $$

- Método *complete*:

$$ {}^5(C_3, C_5) \succ {}^5(C_2, C_5) \succ {}^5(C_1, C_2) \succ {}^5(C_1, C_4) \succ {}^5(C_1, C_5) \sim {}^5(C_4, C_5) \succ {}^5(C_3, C_4) \succ {}^5(C_2, C_3) \succ {}^5(C_2, C_4) \succ {}^5(C_1, C_3) $$

Al aplicar Borda, obtenemos que aquí uniremos los clústers: ${}^5(C_3, C_5)$. Esta es la pareja ganadora en ambos rankings y por lo tanto estará en la primera posición del ranking obtenido con el recuento de Borda al tener la máxima puntuación.

Nivel 4 Volvemos a renombrar nuestros clústers:

${}^5C_1 \to {}^4C_1$	${}^5C_4 \to {}^4C_4$
${}^5C_2 \to {}^4C_2$	${}^5C_5 \to {}^4C_3$
${}^5C_3 \to {}^4C_3$	

- Método *single*:

$${}^4(C_1, C_4) \succ {}^4(C_1, C_2) \sim {}^4(C_2, C_3) \succ {}^4(C_1, C_3) \succ {}^4(C_3, C_4) \succ {}^4(C_2, C_4)$$

- Método *complete*:

$${}^4(C_1, C_2) \succ {}^4(C_1, C_4) \succ {}^4(C_3, C_4) \succ {}^4(C_2, C_3) \succ {}^4(C_2, C_4) \succ {}^4(C_1, C_3)$$

Aplicamos el recuento Borda a nuestro perfil de rankings. Este proceso se detalla en la tabla mostrada a continuación ya que en este caso no es tan inmediato ver el ganador como en el paso anterior:

Ranking	${}^4(C_1, C_4)$	${}^4(C_1, C_2)$	${}^4(C_2, C_3)$	${}^4(C_1, C_3)$	${}^4(C_3, C_4)$	${}^4(C_2, C_4)$
r_{Single}	5	3.5	3.5	2	1	0
r_{Complete}	4	5	2	0	3	1
Total	9	8.5	5.5	2	4	1
Ranking Final	${}^4(C_1, C_4) \succ {}^4(C_1, C_2) \succ {}^4(C_2, C_3) \succ {}^4(C_3, C_4) \succ {}^4(C_3, C_4) \succ {}^4(C_2, C_4)$					

Por tanto el par de clústers a unir será ${}^4(C_1, C_4)$.

Nivel 3 Renombramos los clústers de forma análoga a lo hecho desde el principio. En este caso todas las apariciones de C_4 son sustituidas por C_1.

- Método *single*:

$${}^3(C_1, C_2) \sim {}^3(C_2, C_3) \succ {}^3(C_1, C_3)$$

- Método *complete*:

$${}^3(C_2, C_3) \succ {}^3(C_1, C_2) \succ {}^3(C_1, C_3)$$

Realizamos de nuevo el recuento Borda, ahora mucho más simple al considerar 3 parejas únicamente:

Ranking	${}^3(C_1, C_2)$	${}^3(C_2, C_3)$	${}^3(C_1, C_3)$
r_{Single}	1.5	1.5	0
r_{Complete}	1	2	0
Total	2.5	3.5	0
Ranking Final	${}^3(C_2, C_3) \succ {}^3(C_1, C_2) \succ {}^3(C_1, C_3)$		

Por tanto escogemos ${}^{3}(C_2, C_3)$, al ser la primera opción de ambos criterios.

Nivel 2 Renombramos y obtenemos un ranking de un único elemento.

- Método *single* y *complete*:

$$^{2}(C_1, C_2)$$

Optamos por la única opción para la última iteración, que siempre será inmediata y nunca hará falta programar el recuento.

Nivel 1 Nos queda un único clúster final llamado ${}^{1}C_1$.

4.2.2. Datos reales

Además de probar y desarrollar nuestro algoritmo en un conjunto de datos de muestra, hemos querido ir más allá y emplearlo en el conjunto de datos reales mostrado en la Figura 4.2.

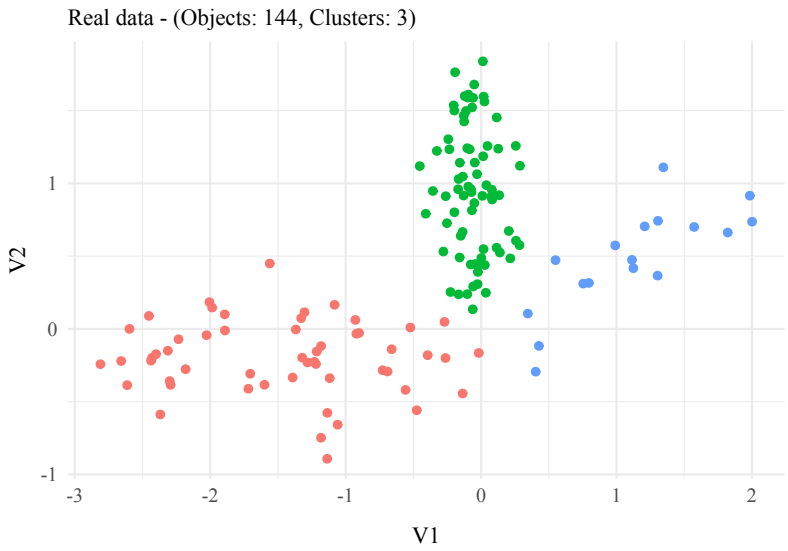

Figura 4.2: Representación gráfica del conjunto de datos reales sobre el que aplicamos nuestro algoritmo.

Se procede a comparar tres resultados del algoritmos clásico utilizando la distancia Euclídea y en cada uno de los tres casos utilizando un criterio de adhesión diferente: *single*, *complete* o *average*.

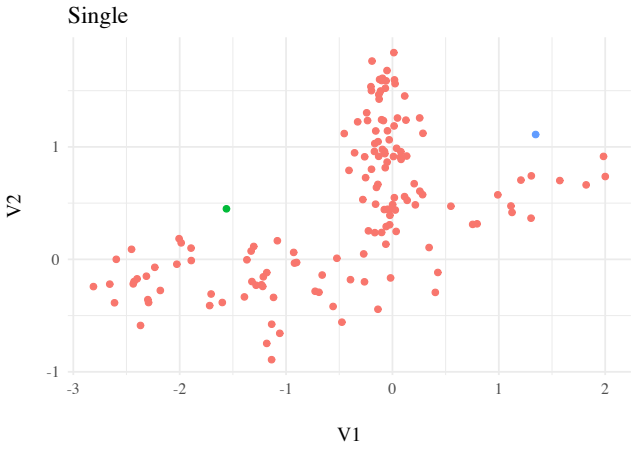

Figura 4.3: Representación gráfica de la agrupación en tres clústers de datos reales empleando el criterio *single*.

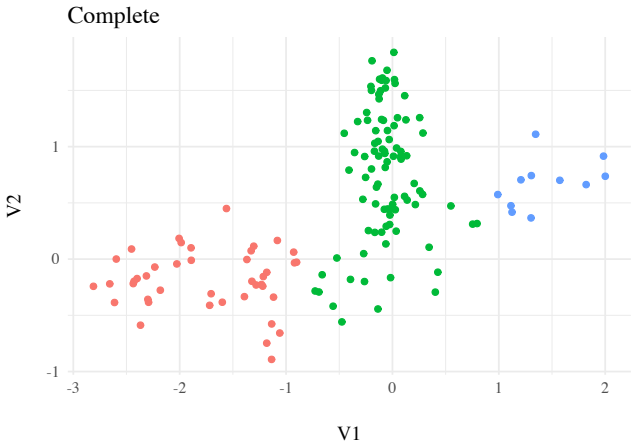

Figura 4.4: Representación gráfica de la agrupación en tres clústers de datos reales empleando el criterio *complete*.

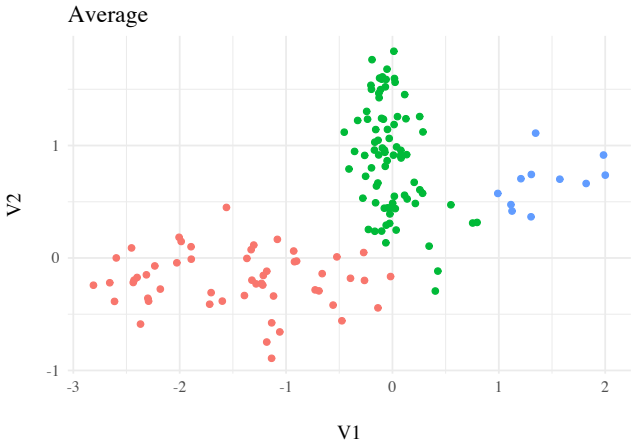

Figura 4.5: Representación gráfica de la agrupación en tres clústers de datos reales empleando el criterio *average*.

Estos resultados individuales se comparan con los resultados del método propuesto utilizando la distancia Euclídea y la agregación de los tres criterios con el recuento Borda.

Por último, en la Figura 4.6 se observa como los resultados obtenidos con el método de propuesto coinciden con los resultados del conjunto original, mientras que los resultados obtenidos con el algoritmo clásico son muy diversos dependiendo del criterio de agregación utilizado.

Gracias a este ejemplo con datos reales, podemos comprobar, que aunque en un primer momento se desecharía el empleo del método *single* por no ser nada útil en la representación de los datos, cuando se le introduce en un criterio de agrupación que hace uso de la agregación de rankings podemos lograr resultados determinantes.

El consenso entre distintos criterios de adhesión es la clave de este trabajo, pero a la hora de su implementación he podido encontrar diferentes dificultades. En la siguiente Sección, vamos a poder ver una de ellas.

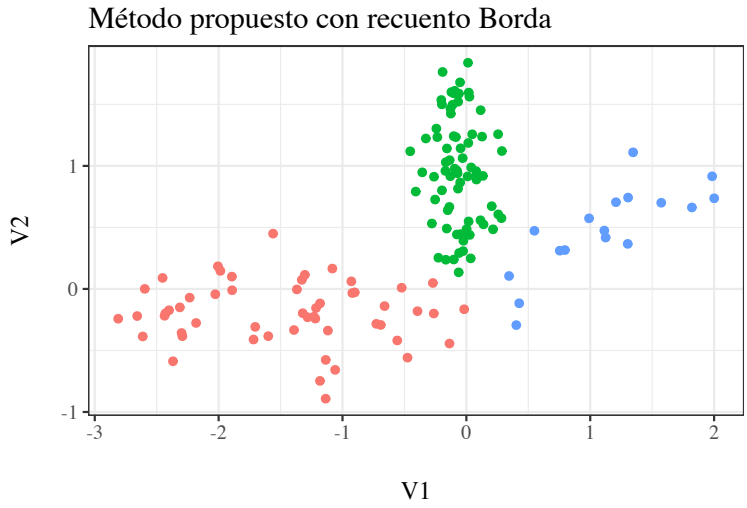

Figura 4.6: Representación gráfica de la agrupación en tres clústers de datos reales empleando la agregación de rankings mediante el recuento Borda a partir del criterio *single*, *complete* y *average*.

4.3. Resolución de empates

Al realizar la agregación de rankings con el recuento Borda, puede ocurrir que haya más de un par de candidatos que obtenga la máxima puntuación y por lo tanto más de un par esté empatado como ganador del ranking. Esto puede ocurrir también en el algoritmo original, ya que al utilizar la distancia Euclídea puede haber más de un par de objetos que estén a la misma distancia y esa distancia sea la mínima. No obstante, en esta sección se proponen métodos de resolución de empates para lidiar con esta situación en la medida de lo posible en el algoritmo propuesto.

Gracias a estos vamos a encontrar las diferencias entre los criterios de adhesión *single* y *complete*, y desempatando podemos llegar a un balance entre ambos.

A continuación se ofrecen diferentes técnicas para desempatar, que desarrollaremos en la siguiente sección:

1. Hacer prevalecer conjuntos unipuntuales.
2. Aplicar Borda ponderado.
3. Aplicar Borda invertido.

4.4. Empates en el recuento Borda: cómo solucionarlos

Estas propuestas buscan dar solución a la situación donde hay más de un par de candidatos empatados en la primera posición del ranking ganador resultante de aplicar el recuento Borda. Si bien pueden surgir empates entre otros candidatos, como se ha visto en el Ejemplo 3.3, estos no suponen un problema en la ejecución del algoritmo.

4.4.1. Desempate al azar

A pesar de ser una solución infundada, es cierto que en muchas implementaciones individuales de diferentes criterios de agregación los desempates se resuelven al azar. Esto se realiza ya que, si los objetos están definidos como vectores en $\mathbb{R}$, podemos encontrarnos con objetos exactamente iguales en el mismo set de datos u objetos simétricos que siempre llevarían a empate, para los que no habría ningún criterio coherente de desempate. Se presupone que trabajando con conjuntos de datos reales, con alta dimensionalidad, estás situaciones no serán frecuentes y por lo tanto se implementa un desempate al azar.

No obstante, a la hora de nuestra propuesta de método, intentando buscar un consenso entre distintos criterios de adhesión a través del recuento Borda, es común encontrarnos con empates dentro de la agregación de rankings.

Implementar un desempate al azar para que se realice de manera continua puede afectar a la homogeneidad de las posteriores subpoblaciones extraídas,

sin entrar a mencionar que para que sea verdaderamente aleatoria la elección del candidato ganador, deberíamos programarlo específicamente de esta manera en nuestro algoritmo.

Por ejemplo, en el caso del lenguaje de programación `R`, si dejamos escoger al azar nuestra pareja de clústers ganadora, normalmente colocará en primera posición aquella que haya sido leída o almacenada primero[2].

En el algoritmo propuesto, el desempate al azar debe ser evitado y únicamente se lleva a cabo en el caso de no ser posible desempatar con ninguno de los otros criterios propuestos.

4.4.2. Hacer prevalecer clústers unipuntuales

La idea detrás de esta propuesta es que, en cuanto sea posible, se deberán priorizar aquellas parejas que contengan clústers unipuntuales. Si no se puede, podremos optar por uno de los otros dos métodos siguientes.

La justificación detrás de esta acción surge exclusivamente del afán de no dejar conjuntos unipuntuales de cara al estudio posterior. El analista es el encargado de escoger los clústers idóneos y considerarlos subpoblaciones, para así hacer una interpretación correcta de los datos. Cabe destacar que la existencia de subpoblaciones unipuntuales no nos permite agrupar correctamente nuestro conjunto de objetos, y que deberían intentar evitarse siempre que el conjunto lo permita.

Dentro de este criterio de desempate, se priorizará la unión de dos clústers unipuntuales a la unión de un clúster unipuntual a otro de mayor cardinal.

Este criterio se puede extrapolar a cualquier clúster, de manera que los clústers de mayor tamaño *cedan el paso* en la iteración a aquellos de menor

[2]Para lograr que de verdad sea totalmente aleatoria, se tendría que instruir a nuestro algoritmo que dentro del set de candidatos en primera posición, escoja uno al azar, no que simplemente escoja aquel que a la hora de almacenar los elementos en primera posición del ranking agregado resultante esté colocado el primero en orden de lectura.

tamaño. Esta extrapolación debe hacerse con sumo cuidado, pues podríamos llegar a tener grupos poco homogéneos.

4.4.3. Borda ponderado

Para esta solución se propone introducir el concepto de *coeficiente de ponderación adaptado.*

Definición 4.1. *El* ***coeficiente de ponderación adaptado*** $\alpha^j_{A_i,B_i}$*, para una pareja de clústers* $(C_{A_i}, C_{B_i}), i \in \{1, \ldots, L_\nu\}$ *en el ranking* $r_j \equiv (C_{A_1}, C_{B_1}) \succ (C_{A_2}, C_{B_2}) \succ (C_{A_3}, C_{B_3}) \succ (C_{A_4}, C_{B_4}) \succ \cdots \succ (C_{A_{L_\nu}}, C_{B_{L_\nu}})$ *de un perfil de rankings de* M *rankings, se define como:*

$$\alpha^j_{A_i,B_i} = \begin{cases} 0 & si\ i = L_\nu \\ \frac{1}{M}\frac{d_j(C_{A_i},C_{B_i})}{d_j(C_{A_{i+1}},C_{B_{i+1}})} & en\ cualquier\ otro\ caso \end{cases} \tag{4.1}$$

Partiendo de este coeficiente de ponderación adaptado de cada ranking r_j podemos calcular un coeficiente de ponderación que nos ayude a resolver los empates. Su fórmula es la siguiente, para una pareja (C_A, C_B) en un perfil de ránkings de M rankings:

$$\alpha_{A,B} = \alpha^M_{A,B} + \alpha^{M-1}_{A,B} + \cdots + \alpha^1_{A,B}$$

Debemos notar que aunque se podría hacer un Borda ponderado para todos los elementos empatados (o desde el principio), esta idea debería descartarse ya que su coste computacional ralentizaría la ejecución del algoritmo.

No obstante, en caso de que se quisiera hacer un recuento Borda ponderado desde un principio, es posible hacerlo con la definición del coeficiente de ponderación adaptado, que se denomina así por estar dividido entre el número de rankings contenido en el perfil. La razón detrás de esta corrección es para que el coeficiente de ponderación $\alpha_{A,B}$ esté en el intervalo $[0, 1]$.

Si no se diseña de esta manera, y aplicásemos el recuento Borda ponderado a todo el ranking, podría haber *adelantos* por parte de algunas parejas

que estén muy próximas entre sí en el caso hipotético de que consideremos todas las parejas. Esto es debido a este tipo de ponderación *castiga* en exceso aquellos candidatos que aparecen alguna vez al final de un ranking, otorgando directamente una puntuación de cero y un coeficiente de ponderación adaptado de cero también.

Es más, aunque no estemos aplicándolo a todo el ranking y solo nos enfoquemos en los empates en primera posición, si uno de esos candidatos empatados figura en última posición en algún ranking, sus posibilidades de salir como candidato final del desempate se vuelven muy pequeñas.

Aún así todavía podríamos seguir teniendo empates, sobre todo si hay empates dentro de los rankings del perfil, por lo que proponemos otro método más.

4.4.4. Borda invertido

Una ventaja de esta solución en comparación con la anterior es que es fácil de implementar: se basa en sumar a los elementos empatados otra vez su puntuación, pero de manera invertida. Esto refuerza a aquellos elementos que tienen las posiciones más elevadas en los rankings del perfil. Si la puntuación de la pareja (C_A, C_B) en el ranking r_j de tamaño n del perfil de rankings es $p^j_{A,B}$, su puntuación asignada en con la modificación del recuento Borda invertido será:

$$^{-1}p^j_{A,B} = p^j_{A,B} + \frac{1}{n - p^j_{A,B}}$$

Este método puede solucionar algunos empates de manera muy rápida, cuando las puntuaciones que están empatadas en la agregación de rankings no están conformadas por los mismos sumandos.

Lo positivo de esta manera de desempatar, en comparación con las anteriores, es que vuelve a recurrir al método Borda, aunque de manera modificada. No recurrir a las distancias proporciona a este método otra ventaja: se trata de un método puramente matemático cuya carga computacional es

menor y que se puede aplicar a otro tipo de desempates del recuento Borda, no relacionados con distancias y parejas de clústers.

No obstante, aplicando el Borda invertido no siempre llegamos a resolver empates, ya que en el caso del ejemplo de la Sección 4.5 llegaremos a un empate donde este método en concreto no nos será de ayuda.

Después de esta serie de propuestas de desempate, vamos a *reciclar* un ejemplo ya visto para ver su funcionamiento.

4.5. Ejemplo con empates del recuento Borda aplicado al *clustering* jerárquico ascendente

Recuperando el ejemplo utilizado en la Sección 2.2, pondremos en práctica las metodologías de desempate vistas. Para ello, nos centraremos exclusivamente el criterio *single* y *complete*.

Nivel 6 Como en el ejemplo detallado anteriormente, solamente realizamos un ranking general de los clústers:

$$^6(C_4, C_5) \succ {}^6(C_4, C_6) \succ {}^6(C_3, C_4) \succ {}^6(C_2, C_5) \succ {}^6(C_1, C_2) \succ$$
$$^6(C_2, C_3) \succ {}^6(C_5, C_6) \succ {}^6(C_1, C_5) \succ {}^6(C_2, C_4) \succ {}^6(C_3, C_5) \succ$$
$$^6(C_3, C_6) \succ {}^6(C_1, C_4) \succ {}^6(C_2, C_6) \succ {}^6(C_1, C_3) \succ {}^6(C_1, C_6)$$

Aquí claramente vemos que los clústers a unir son: $^6(C_4, C_5)$.

Nivel 5 Renombramos nuestros clústers:

$^6C_1 \to {}^5C_1$	$^6C_4 \to {}^5C_4$
$^6C_2 \to {}^5C_2$	$^6C_5 \to {}^5C_4$
$^6C_3 \to {}^5C_3$	$^6C_6 \to {}^5C_5$

Para no ser repetitivos, vamos a condensar nuestros renombramientos en la matriz (4.4), diseñada específicamente para ello, al igual que el primer ejemplo. Hagamos los rankings según los dos métodos.

- Método *single*:

$$^5(C_4, C_5) \succ ^5(C_3, C_4) \succ ^5(C_2, C_4) \succ ^5(C_1, C_2) \succ ^5(C_2, C_3) \succ$$
$$^5(C_1, C_4) \succ ^5(C_3, C_5) \succ ^5(C_2, C_5) \succ ^5(C_1, C_3) \succ ^5(C_1, C_5)$$

- Método *complete*:

$$^5(C_1, C_2) \succ ^5(C_2, C_3) \succ ^5(C_4, C_5) \succ ^5(C_2, C_4) \succ ^5(C_3, C_4) \succ$$
$$^5(C_3, C_5) \succ ^5(C_1, C_4) \succ ^5(C_2, C_5) \succ ^5(C_1, C_3) \succ ^5(C_1, C_5)$$

Como lo que nos interesa en este ejemplo es la resolución de empates, solo presentaremos el perfil de rankings de los 5 primeros elementos. No obstante, en (4.2) encontramos el resultado del recuento Borda para todos los elementos:

Ranking	$^5(C_4, C_5)$	$^5(C_3, C_4)$	$^5(C_2, C_4)$	$^5(C_1, C_2)$	$^5(C_2, C_3)$
r_{Single}	9	8	7	6	5
r_{Complete}	7	5	6	9	8
Total	16	13	13	15	13
Ranking Final	$^5(C_4, C_5) \succ ^5(C_1, C_2) \succ ^5(C_2, C_4) \sim$ $^5(C_2, C_3) \sim ^5(C_3, C_4)$				

$$\begin{aligned} &^5(C_4, C_5) \succ ^5(C_1, C_2) \succ ^5(C_2, C_4) \sim ^5(C_2, C_3) \sim \\ &^5(C_3, C_4) \succ ^5(C_2, C_4) \sim ^5(C_3, C_5) \succ ^5(C_2, C_5) \succ \\ &^5(C_1, C_3) \succ ^5(C_2, C_6) \end{aligned} \tag{4.2}$$

Ahora nuestro par de clústers a unir será $^5(C_4, C_5)$.

Nivel 4 Vamos a volver a renombrar nuestros clústers y a reinterpretar las distancias.

- Método *single*:

$$^4(C_3, C_4) \succ ^4(C_2, C_4) \succ ^4(C_1, C_2) \succ ^4(C_2, C_3) \succ ^4(C_1, C_4) \succ$$
$$^4(C_1, C_3)$$

- Método *complete*:

$$^4(C_1, C_2) \succ ^4(C_2, C_3) \succ ^4(C_3, C_4) \succ ^4(C_2, C_4) \succ ^4(C_1, C_3) \succ$$
$$^4(C_1, C_4)$$

En recuento Borda nos dará como resultado:

Ranking	${}^4(C_3, C_4)$	${}^4(C_2, C_4)$	${}^4(C_1, C_2)$	${}^4(C_2, C_3)$
r_{Single}	5	4	3	2
r_{Complete}	3	2	5	4
Total	8	6	8	6
Ranking Final	${}^4(C_1, C_2) \sim {}^4(C_3, C_4) \succ {}^4(C_2, C_3) \sim {}^4(C_2, C_4)$			

De manera general sería:

$$ {}^4(C_1, C_2) \sim {}^4(C_3, C_4) \succ {}^4(C_2, C_3) \sim {}^4(C_2, C_4) \succ {}^4(C_1, C_3) \sim {}^4(C_1, C_4) $$

Ahora que hemos llegado a un empate, podemos aplicar diferentes métodos para solucionarlo:

1. **Borda ponderado**: Como habíamos mencionado con anterioridad, en nuestra propuesta el recuento Borda ponderado parte exclusivamente de nuestro recuento Borda sin ponderar, y solo estudia los coeficientes de ponderación adaptado (4.1) de aquellas parejas de clústers que hayan quedado empatadas en el ranking agregado resultante. En nuestro caso tenemos que empatan los parejas ${}^4(C_1, C_2)$ y ${}^4(C_3, C_4)$.

   Así tenemos que retomar los dos rankings del perfil, r_{Single} y r_{Complete}, y buscar las apariciones de ${}^4(C_1, C_2)$ y ${}^4(C_3, C_4)$. Ahora, tomaremos las parejas de clústers anteriores a esas apariciones, y realizaremos un total de cuatro cocientes (obteniendo así cuatro coeficientes de ponderación adaptados).

   En la siguiente Tabla, podemos ver el proceso detrás de los cálculos de los cuatro coeficientes de ponderación adaptados. En cada columna, encontramos las distancias entre las parejas empatadas del ranking agregado y las distancias entre las parejas anteriores a estas en cada respectivo ranking del perfil.

r_{Single}	r_{Complete}
$d_{\text{Single}}({}^4C_3, {}^4C_4) =$ $d({}^6C_3, {}^6C_4) = 1.874365$	$d_{\text{Complete}}({}^4C_3, {}^4C_4) =$ $d({}^6C_3, {}^6C_6) = 3.160455$
$d_{\text{Single}}({}^4C_2, {}^4C_4) =$ $d({}^6C_2, {}^6C_4) = 2.024253$	$d_{\text{Complete}}({}^4C_2, {}^4C_4) =$ $d({}^6C_2, {}^6C_6) = 3.564772$
$\alpha_{3,4}^{\text{Single}} = \frac{1.874365}{2 \cdot 2.024253} =$ 0.462977	$\alpha_{3,4}^{\text{Complete}} = \frac{3.160455}{2 \cdot 3.564772} =$ 0.443290
$d_{\text{Single}}({}^4C_1, {}^4C_2) =$ $d({}^6C_1, {}^6C_2) = 2.058543$	$d_{\text{Complete}}({}^4C_1, {}^4C_2) =$ $d({}^6C_1, {}^6C_2) = 2.058543$
$d_{\text{Single}}({}^4C_2, {}^4C_3) =$ $d({}^6C_2, {}^6C_3) = 2.065736$	$d_{\text{Complete}}({}^4C_2, {}^4C_3) =$ $d({}^6C_2, {}^6C_3) = 2.065736$
$\alpha_{1,2}^{\text{Single}} = \frac{2.058543}{2 \cdot 2.065736} =$ 0.498259	$\alpha_{1,2}^{\text{Complete}} = \frac{2.058543}{2 \cdot 2.065736} =$ 0.498259

Así nos quedaría el perfil de rankings, añadiendo los coeficientes de ponderación adaptados calculados [3]:

Ranking	${}^4(C_3, C_4)$	${}^4(C_2, C_4)$	${}^4(C_1, C_2)$	${}^4(C_2, C_3)$
r^p_{Single}	5.462977	4	3.498259	2
r^p_{Complete}	3.443290	2	5.498259	4
Total	8.906267	6	8.9980849	6
Ranking Final	${}^4(C_1, C_2) \succ {}^4(C_3, C_4) \succ {}^4(C_3, C_4) \sim {}^4(C_2, C_4)$			

2. **Prevalencia de clústers unipuntuales**: ${}^4C_4$ está conformado por tres elementos, mientras que ${}^4C_1$, ${}^4C_2$ y ${}^4C_3$ son unipuntuales.
3. **Al azar escogido por** `R`: `R` sitúa siempre a la izquierda aquellos clúster con el índice más pequeño.
4. **Borda invertido**: este es un posible ejemplo donde este método de desempate no nos es de utilidad.

   Esto se debe a que ambos candidatos en primera posición obtienen la misma puntuación a raíz de los mismos sumandos $(5 + 3)$, de

[3] Si quisiésemos, podíamos calcular los coeficientes de ponderación directamente, sin volver a nuestra tabla, y escoger directamente la pareja cuyo coeficiente sea mayor, es exactamente lo mismo.

manera que su puntuación final bajo Borda invertido será:

$$8 + \frac{1}{6-5} + \frac{1}{6-3} = 8 + 1 + \frac{1}{3} = 9.333\ldots$$

Obviando el Borda invertido ya que no nos aporta una solución, da igual qué método escojamos para solucionar el perfil rankings. Los otros tres métodos restantes en este caso proponen exactamente lo mismo: ${}^{4}(C_1, C_2)$

Nivel 3 Renombramos nuestros clústers como hemos hecho desde el principio y preparamos la penúltima iteración.

- Método *single*:

$${}^{3}(C_2, C_3) \succ {}^{3}(C_1, C_3) \succ {}^{3}(C_1, C_2)$$

- Método *complete*:

$${}^{3}(C_2, C_3) \succ {}^{3}(C_1, C_2) \succ {}^{3}(C_1, C_3)$$

Ahora, escogemos ${}^{3}(C_2, C_3)$, al ser la primera opción de ambos criterios.

Nivel 2

- Método *single* y *complete*:

$${}^{2}(C_1, C_2)$$

Hemos llegado a la última iteración, inmediata.

Nivel 1 Nos queda un único clúster final llamado ${}^{1}C_1$.

4.6. Interpretación de los resultados del algoritmo

A continuación introduciremos el concepto de *matriz de renombramientos*. Para la representación de los resultados del análisis clúster bajo el recuento Borda no se puede usar un dendrograma, ya que la hora de construirlo las alturas entre los nodos simbolizan la diferencia de distancia a la que están. Esta diferencia no está disponible en la representación en rankings, y además,

si estuviera, como estamos trabajando con diferentes criterios de agregación, cada uno de estos denota de manera distinta la distancia entre dos clústers, haciendo inviable escoger una sobre otra por los problemas de representación que nos podrían surgir.

Sin embargo, sí podemos representar nuestras uniones en una matriz indicando en cada nivel qué elementos están en el mismo clúster.

Definición 4.2. *Sea una matriz $A \in \mathcal{M}_n(\{1, 2, \ldots, n\})$. Decimos que A es la* ***matriz de renombramientos*** *asociada a la clusterización jerárquica ascendente mediante el recuento Borda del conjunto $\mathscr{C}$, cuando en cada fila $A_{i\cdot}$ se representa la disposición de los elementos del conjunto en sus respectivos clústers para cada nivel $i \in \{1, \ldots, n\}$.*

Ejemplo 4.1. *Para el ejemplo de la Sección 4.2, su matriz de renombramientos sería*

$$\begin{array}{ll} \textit{Nivel} & \textit{Elementos} \\ 1 \to & \\ 2 \to & \\ 3 \to & \\ 4 \to & \\ 5 \to & \\ 6 \to & \end{array} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 2 & 1 & 3 & 1 & 3 \\ 1 & 2 & 1 & 3 & 4 & 3 \\ 1 & 2 & 1 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix} = A \tag{4.3}$$

Análogamente, para el ejemplo de la Sección 4.5, tenemos la siguiente matriz de renombramientos:

$$\begin{array}{ll} \textit{Nivel} & \textit{Elementos} \\ 1 \to & \\ 2 \to & \\ 3 \to & \\ 4 \to & \\ 5 \to & \\ 6 \to & \end{array} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 2 & 2 & 2 & 2 \\ 1 & 1 & 2 & 3 & 3 & 3 \\ 1 & 2 & 3 & 4 & 4 & 4 \\ 1 & 2 & 3 & 4 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix} = B \tag{4.4}$$

Este tipo de matriz se denomina ***matriz de renombramientos rigurosa****, ya que, en cada nivel; al disminuir en uno el número de elementos, se corrigen los índices de los clústers en una unidad. Esto otorga nitidez al ejemplo, pero a veces puede ser difícil de interpretar, pues en cada nivel varían varios números cuando en realidad solo se están uniendo dos clústers, además de que dificulta la identificación de clústers unipuntuales a unir cuando tienen índices altos.*

Por esta precisa razón, introducimos el concepto de ***matriz de renombramientos nivelada****, donde en cada nivel solo se modifica un índice con respecto al nivel anterior. Este tipo de matriz, además de comprenderse de manera más inmediata, es más fácil de implementar en un programa informático, a la vez que más rápida.*

Veamos como quedaría la matriz de renombramientos rigurosa anterior (4.4) *bajo este nuevo concepto de matriz:*

$$\begin{array}{ll} \textit{Nivel} & \textit{Elementos} \\ 1 \rightarrow & \\ 2 \rightarrow & \\ 3 \rightarrow & \\ 4 \rightarrow & \\ 5 \rightarrow & \\ 6 \rightarrow & \end{array} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 3 & 3 & 3 & 3 \\ 1 & 1 & 3 & 4 & 4 & 4 \\ 1 & 2 & 3 & 4 & 4 & 4 \\ 1 & 2 & 3 & 4 & 4 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix} = B$$

Es gracias a las matrices de renombramientos (rigurosas en este caso), que vemos que obtenemos resultados distintos que el método *single* o *complete*[4] del ejemplo visto recientemente en la Sección 4.5:

Nivel	Elementos	Nivel	Elementos
$1 \rightarrow$	$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$	$1 \rightarrow$	$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$
$2 \rightarrow$	$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 2 \end{pmatrix}$	$2 \rightarrow$	$\begin{pmatrix} 1 & 1 & 2 & 2 & 2 & 2 \end{pmatrix}$
$3 \rightarrow$	$\begin{pmatrix} 1 & 1 & 2 & 2 & 2 & 3 \end{pmatrix}$	$3 \rightarrow$	$\begin{pmatrix} 1 & 1 & 2 & 3 & 3 & 3 \end{pmatrix}$
$4 \rightarrow$	$\begin{pmatrix} 1 & 2 & 3 & 3 & 3 & 4 \end{pmatrix}$	$4 \rightarrow$	$\begin{pmatrix} 1 & 1 & 2 & 3 & 3 & 4 \end{pmatrix}$
$5 \rightarrow$	$\begin{pmatrix} 1 & 2 & 3 & 4 & 4 & 5 \end{pmatrix}$	$5 \rightarrow$	$\begin{pmatrix} 1 & 2 & 3 & 4 & 4 & 5 \end{pmatrix}$
$6 \rightarrow$	$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix}$	$6 \rightarrow$	$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix}$
Método *single*		Método *complete*	

De hecho, para visualizar los resultados, podemos recurrir a un representación cromática. Para ello, se ha diseñado una manera de visualizar el contenido de la matriz de renombramientos a través de una serie de colores.

Para el ejemplo con empates, esta es su representación, para cada uno de los métodos de agregación. Vamos a denominarlo *plagrama* (de πλακάς «plakás», azulejo en griego antiguo) o *tilegrams*, del inglés.

[4]Las matrices de renombramientos se deducen de manera inmediata de los dendrogramas 2.2 y 2.3 respectivamente.

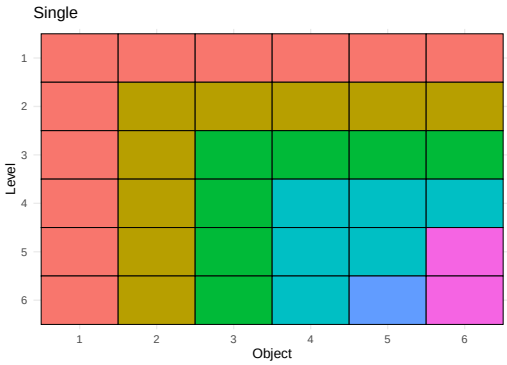

Figura 4.7: Plagrama para la agrupación bajo el criterio *single* de la Figura 2.1

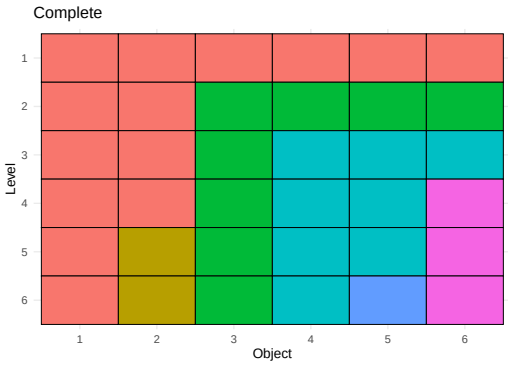

Figura 4.8: Plagrama para la agrupación bajo el criterio *complete* de la Figura 2.1

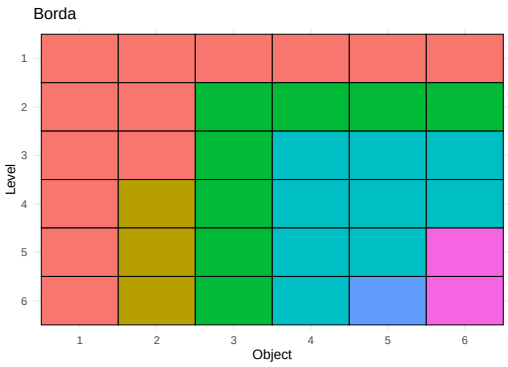

Figura 4.9: Plagrama para la agrupación agregada con el recuento Borda de la Figura 2.1

De esta forma es sencillo ilustrar cómo los resultados obtenidos son diferentes. Podemos hacer lo mismo para el ejemplo de la Sección 4.2:

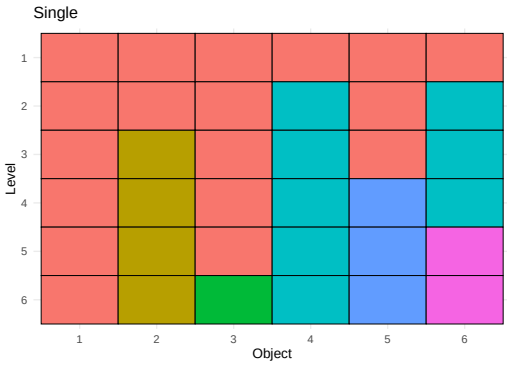

Figura 4.10: Plagrama para la agrupación bajo el criterio *single* de la Figura 4.1

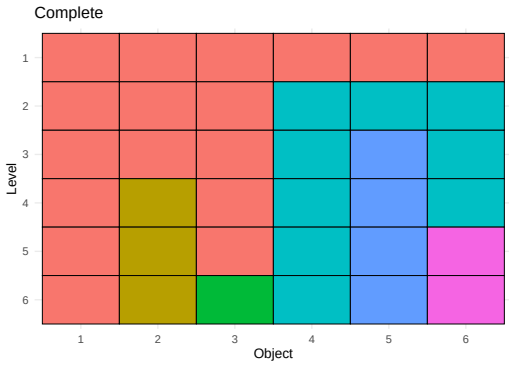

Figura 4.11: Plagrama para la agrupación bajo el criterio *complete* de la Figura 4.1

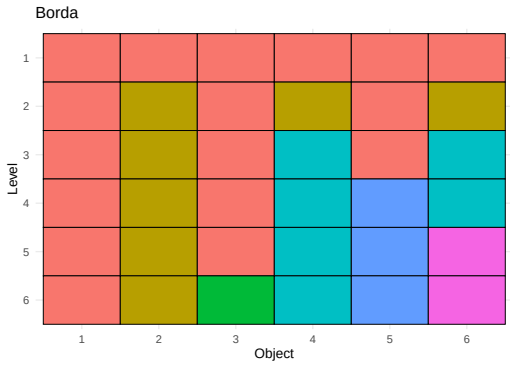

Figura 4.12: Plagrama para la agrupación agregada con el recuento Borda de la Figura 4.1

Capítulo 5

Conclusiones

En este capítulo se presentan las conclusiones finales tras realizar este trabajo. Además se proponen futuras líneas de investigación tomando como base el algoritmo propuesto.

5.1. Valoración final del trabajo

Tras la realización del trabajo, se han logrado los objetivos propuestos al comienzo de su desarrollo. Caben destacar los siguientes:

- Investigar sobre diferentes criterios de agregación, viendo las desventajas de cada uno de ellos y haciendo una propuesta que suavice estas.
- Crear una agregación de rankings específicamente diseñada para la agrupación jerárquica ascendente, prestando atención a las peculiaridades del método.
- Obtener resultados distintos a los que emplean únicamente métodos de agregación uno a uno.
- Proponer una representación visual de los resultados obtenidos.

En lo respectivo al ámbito personal, me gustaría mencionar que gracias a la elaboración de este trabajo he tenido oportunidad de ampliar mi conocimiento en áreas relacionadas con el análisis, tratamiento y visualización de datos.

Los diversos retos surgidos en el transcurso, concretamente a la hora de buscar técnicas de desempate para el recuento Borda, me han permitido iniciarme en la investigación matemática.

Por otra parte, me llevo conmigo una experiencia que en un futuro me podrá permitir enfrentarme a proyectos de mayor envergadura, siendo consciente de la importancia de su planificación.

Me gustaría concluir señalando lo fructífero, tanto personal como académicamente, que ha sido poder realizar un Trabajo Fin de Estudios en este campo de las Matemáticas. En el Análisis de Datos, la incertidumbre y la sorpresa están a la orden del día: hay numerosos ejemplos, llenos de curiosidades y peculiaridades con los que podido trabajar. Y son precisamente estos ejemplos los que han ido desafiando mi paciencia, forzándome a perfilar el pseudocódigo de mi propuesta de algoritmo y su posterior programación, para que se ajuste a todo tipo de casos y excepciones posibles (como hemos podido ver con los empates en la agregación de rankings).

5.2. Futuras líneas de investigación

A continuación se proponen futuras líneas de investigación que pueden derivarse del algoritmo propuesto en este trabajo.

Comparativa de diferentes métodos de agrupación a la agregación de rankings. En esta memoria nos hemos limitado exclusivamente a la comparación del algoritmo propuesto con los clásicos utilizando el criterio *single* y *complete* en nuestros ejemplos de agregación de rankings. El algoritmo propuesto es extensible a todas las metodologías estudiadas y por lo tanto sería interesante estudiar su comportamiento.

Recuento Borda en el *clustering* jerárquico ascendente aplicado a similitudes. La aplicación de este trabajo se ha realizado empleando siempre la distancia Euclídea. Sin embargo, el algoritmo propuesto es válido para cualquier otra métrica de similitud entre objetos. Sería interesante realizar un estudio exhaustivo sobre el comportamiento del algoritmo propuesto cuando varía esta métrica.

Representación de resultados de la agrupación en agregación de rankings a través de dendrogramas. La altura de los dendrogramas representa la distancia a la que se ha unido esa pareja de clústers. En nuestra propuesta de algoritmo, no podemos representar directamente nuestros resultados en dendrogramas, puesto que dos clústers no están a una distancia fija. ¿Cómo podemos solventar este problema? Una futura línea de investigación pasaría por diseñar una representación similar al dendrograma para métodos de agrupación basados en la agregación.

Bibliografía

[1] D. Xu and Y. Tian, "A comprehensive survey of clustering algorithms," *Ann. Data Sci.*, vol. 2, no. 2, pp. 165–193, 2015.

[2] P. H. A. Sneath, "The application of computers to taxonomy." *Journal of general microbiology*, vol. 17 1, pp. 201–26, 1957.

[3] T. A. Sorensen, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons," *Biol. Skar.*, vol. 5, pp. 1–34, 1948.

[4] R. Sokal, C. Michener, and U. of Kansas, *A Statistical Method for Evaluating Systematic Relationships*, ser. University of Kansas science bulletin. University of Kansas, 1958. [Online]. Available: https://books.google.es/books?id=o1BlHAAACAAJ

[5] J. C. Borda, *Mémoire sur les Élections au Scrutin.* Paris: Histoire de l'Académie Royale des Sciences, 1781.

[6] C. M. Cuadras, "Distancias estadísticas," *Estadística Española*, no. 119, pp. 295–358, 1988.

[7] B. Sinova Fernández and N. Corral Blanco, "Análisis clúster," Diapositivas de Clase Expositiva de la asignatura Análisis de Datos, 2021.

[8] N. Randriamihamison, N. Vialaneix, and P. Neuvial, "Applicability and interpretability of Ward's hierarchical agglomerative clustering with or without contiguity constraints," *Journal of Classification*, vol. 38, pp. 363–389, 2021. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02294847

[9] J. H. W. Jr., "Hierarchical grouping to optimize an objective function," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 236–244, 1963. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/01621459.1963.10500845

[10] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, *Handbook of Computational Social Choice*, 1st ed. USA: Cambridge University Press, 2016.