

# Agregación (*clustering*)

Ramon Sangüesa i Solé

PID\_00165731



# Índice


<b>Introducción</b> .....	5
<b>Objetivos</b> .....	6
<b>1. Motivación</b> .....	7
<b>2. La similitud, base para la agrupación de objetos</b> .....	8
2.1. Una formulación del problema de la agregación.....	9
<b>3. Espacio, distancia y similitud</b> .....	12
3.1. Medidas de similitud .....	14
3.1.1. Definición y propiedades de las distancias .....	15
<b>4. Métodos de agregación</b> .....	20
4.1. El método de los centroides: <i>k-means</i> .....	20
4.2. El método de los vecinos más cercanos ( <i>k-nearest neighbours</i> ). Proximidad entre grupos .....	21
4.3. Métodos incrementales o aglomeradores.....	22
4.4. Métodos de agregación probabilistas .....	26
4.5. Métodos probabilistas de construcción de agregaciones cuando el número de clases es conocido <i>a priori</i> .....	31
4.6. Métodos de construcción de agregaciones cuando el número de clases es desconocido <i>a priori</i> .....	33
<b>5. Interpretación de los modelos obtenidos</b> .....	35
5.1. Predicción a partir de los métodos de agregación.....	37
5.2. Calidad de los modelos obtenidos .....	40
5.2.1. El principio de mínima longitud de descripción.....	41
5.2.2. Medidas de ajuste .....	45
<b>6. Ponderación de los métodos de agregación</b> .....	48
<b>Resumen</b> .....	49
<b>Actividades</b> .....	51
<b>Ejercicios de autoevaluación</b> .....	51
<b>Bibliografía</b> .....	53



## Introducción

En este módulo presentamos los métodos de agregación\* que dan como resultado los modelos descriptivos. Estos últimos permiten obtener una primera aproximación a la estructura de un dominio y efectuar tareas simples de predicción.

\* En inglés, *clustering*.

Los **métodos de agregación** se utilizan en una situación de relativo desconocimiento del dominio, es decir, de lo que sabemos. Se trata de obtener una descripción inicial que separe grupos de objetos con características parecidas. Esta primera separación debe permitirnos reflexionar acerca de las características comunes de los objetos que pertenecen a cada grupo, lo que los hace parecidos y por qué, y lo que los diferencia de los otros grupos y por qué. 


El tema presenta conceptos importantes que también se utilizan en la obtención de modelos de otros tipos, como son las ideas de **espacio de representación, distancia y medida de similitud**.

## Objetivos

El estudio de los materiales asociados a este módulo permitirá que el estudiante alcance los objetivos siguientes:

- 1.** Conocer los principales métodos de construcción de modelos de agregación.
- 2.** Aprender las nociones básicas utilizadas para definir criterios de proximidad o similitud entre objetos que permiten asegurar que pertenecen a un mismo grupo.
- 3.** Repasar los métodos basados en distancias y en medidas probabilísticas.
- 4.** Conocer sus ámbitos de aplicación más frecuentes.

## 1. Motivación

La empresa Hyper-Gym quiere tener una primera idea de cuáles son sus diferentes grupos de clientes y en qué se parecen entre sí. 

Se trata de encontrar un método que permita discernir, de acuerdo con el contenido de la base de datos, qué clientes son parecidos, pero sin imponer ningún criterio *a priori*. Subrayamos esto: no decimos que los clientes se agrupen porque tengan el mismo atributo (por ejemplo, el horario o la renta), o que conozcamos *a priori* qué etiqueta de clase tiene cada uno, sino que, sin más información que la que aparece en los datos, debemos iniciar un proceso automático cuyo resultado proporcione una división del conjunto original de clientes en subconjuntos formados por clientes que podamos reconocer como parecidos.

Los **métodos de agregación** pretenden encontrar precisamente las clases en que puede dividirse el dominio, el conjunto de observaciones.

## 2. La similitud, base para la agrupación de objetos

La base de la agrupación es tener la capacidad de detectar objetos parecidos. Lo más importante es saber qué aspecto determina que dos objetos sean parecidos.

Se trata de “no mezclar peras y manzanas”, como nos han dicho desde pequeños. Ahora bien, ¿qué factor determina que estos dos tipos de objetos sean considerados diferentes? Después de todo, ambos son frutas, ¿verdad? Sin embargo, debemos centrarnos en los atributos que caracterizan a cada una: color, forma, época de la cosecha, etc. Cada uno de estos atributos puede adoptar un conjunto de valores determinado cuando consideramos todos los objetos que se pueden clasificar. Por ejemplo, el color puede variar entre el verde, el verde claro, el verde ácido, el amarillo, el rojo, el granate, etc.

Debemos fijarnos en que hay varias combinaciones de atributos y valores que nos ayudan a discriminar, que permiten separar un objeto de otro. Por ejemplo, no hay peras rojas ni granates.

Por lo tanto, el objetivo de la agregación consiste en determinar cómo podemos separar un conjunto de objetos en varios grupos a partir de las combinaciones presentes de atributos y valores, de manera que los objetos más parecidos estén en el mismo grupo y los objetos diferentes, en grupos diferentes. !

Los **métodos de agregación** tratan de encontrar criterios generales que permitan realizar la agrupación de objetos independientemente del tipo de objetos (clientes, fruta, coches, etc.) que formen el dominio. Se trata de ofrecer métodos válidos e independientes del dominio.


Hay varios aspectos que debemos considerar: !

1) La primera cuestión es saber en qué nos basamos para indicar que dos objetos son parecidos. Puesto que nos interesa reunir en un mismo grupo los objetos que sean más parecidos entre sí, será preciso que establezcamos alguna manera de medir la similitud entre dos objetos. Los métodos deberán recurrir a las propiedades que podemos observar. En concreto, si observamos dos objetos, tendremos que ver qué combinaciones de atributos y valores muestran y, a partir de ahí, medir su **grado de similitud**.

2) La segunda cuestión es decidir cuándo y cómo colocar un objeto dentro de un grupo de objetos que hemos determinado que eran parecidos: ¿cuando su



similitud con el resto de los objetos sea máxima?, ¿cuando supere un umbral?, ¿cuando se dé una relación de similitud con un porcentaje lo suficiente alto de objetos de la clase? Por ejemplo, si un cliente va al gimnasio por la mañana y tiene una renta alta, ¿en qué clase lo pondremos? ¿En la que corresponde a los clientes que van mayoritariamente al gimnasio por la mañana o en la correspondiente a quienes poseen rentas altas?

Los distintos métodos de agregación de objetos se diferencian entre sí básicamente por su manera de responder a estos problemas. 

### Reflexionad sobre la similitud

Un cliente que va al gimnasio por la mañana y tiene una renta alta, ¿es más parecido a un cliente que va por la tarde y tiene una renta alta o al que va por la mañana y tiene una renta baja?

## 2.1. Una formulación del problema de la agregación

Vamos a plantear el problema de la agregación como un problema de obtención de descripciones a partir de un conjunto de observaciones. Partimos de los elementos siguientes:

a) Un vocabulario de descripción. Las observaciones u objetos que queremos agrupar se describen según un conjunto de atributos determinado. Por ejemplo, en el caso que nos ocupa, cada cliente está descrito en términos de los atributos siguientes: el centro al que está adscrito (*Centro*), el horario en el que está matriculado (*Horario*), la actividad principal que lleva a cabo (*Act1*), la actividad secundaria que realiza (*Act2*), su profesión (*Prof*), su nivel de renta (*Renta*), su edad (*Edad*) y su sexo (*Sexo*).

### El vocabulario...

... determina los atributos que permiten definir observaciones, como por ejemplo, una asignación de valores concretos a cada atributo.

Por lo tanto, cada cliente puede conceptualizarse como una tupla como la siguiente:

(*Centro*, *Horario*, *Act1*, *Act2*, *Prof*, *Renta*, *Edad*, *Sexo*)

Por ejemplo:

(1, 'Mañana', 'Yoga', 'Stretch', 'Jubilado', 3.000.000, 67, 'Hombre')

El número de atributos de cada observación es el mismo: 8. Ello nos permite representar normalmente todo el conjunto de objetos iniciales (clientes, en este caso) como una matriz de objetos y atributos, que también se denomina **matriz de observaciones**. En este caso, cada cliente es un objeto y cada conjunto de valores dado por los diferentes atributos, una observación.

Matriz de observaciones								
Cliente	Centro	Horario	Act1	Act2	Prof	Renta	Edad	Sexo
1	1	Mañana	Yoga	Stretch	Jubilado	3.000.000	68	Mujer
2	3	Tarde	TBC	TBC	Ejecutivo	1.200.000	40	Hombre
3	2	Tarde	Stretch	Steps	Ejecutivo	9.000.000	36	Hombre
4	3	Tarde	TBC	Steps	Ejecutivo	9.000.000	46	Hombre
5	1	Mañana	Aeróbic	Steps	Adm.	4.000.000	30	Mujer

b) Un dominio de valores por atributo. Los atributos pueden tomar valores numéricos (la renta, por ejemplo, que podría oscilar entre un millón y veinticinco millones); o pueden ser categóricos (el tipo de profesión: autónomo, jubilado, funcionario, etc.). En cualquier caso, los métodos actúan sobre una formalización del problema que utiliza una metáfora útil: la del **espacio de atributos** y la de la **similitud** como distancia. Observemos qué quiere decir todo esto.

#### El conjunto de observaciones...

... define el **dominio** de los valores del problema.

Con el fin de agrupar a los clientes de Hyper-Gym, necesitamos definir algún criterio de similitud. ¿Cómo definimos si dos clientes son parecidos? ¿Por qué tienen casi todos los valores iguales? ¿Cómo es esa similitud? ¿Es cero si cumplen que una cierta proporción de atributos son iguales y uno, en caso contrario? ¿O admiten cierta gradación en la similitud?

#### Conceptos importantes

Los conceptos principales para definir la semejanza son el **espacio de atributos** y la **similitud como distancia**.

Es necesario contar con un **criterio de similitud** para agrupar objetos. Más adelante comentaremos algunos.

Podéis ver las medidas de similitud en el subapartado 3.1. de este módulo.



A continuación, presentamos una primera formulación del problema de la agregación.

La **agregación** (*clustering*) es un método que trata de obtener una lista de grupos a partir de una serie de objetos descritos por los valores adoptados por una colección de atributos. La unión de todos los grupos forma el conjunto original de objetos. Un objeto sólo se puede encontrar en un grupo. Los grupos que respetan este criterio se denominan **particiones**.

Hay por lo menos dos formas de describir esta lista de grupos:

- **Descripción extensiva.** Un grupo queda definido por la enumeración de los objetos que pertenecen al mismo. Por ejemplo, supongamos que aplicando cierto método de agregación sobre el conjunto de clientes de Hyper-Gym hemos obtenido una lista de cinco grupos. La descripción del grupo 1 está formada por los clientes que aparecen en éste: (1, 2, 101, 155).
- **Descripción intensiva.** Se hace abstracción de los objetos concretos que forman el grupo y se describen de manera sintética. Por ejemplo, se describe el valor medio de cada atributo para los objetos que están dentro del grupo. Supongamos que hemos obtenido grupos con las características siguientes:

Grupo	Centro	Horario	Act1	Act2	Prof	Renta	Edad	Sexo
1	1	Mañana	Yoga	Stretch	Jubilado	3.000.000	68	Mujer
2	3	Tarde	TBC	Steps	Ejecutivo	7.500.000	32	Hombre
3	1	Mañana	Aeróbic	Steps	Administrativo	3.000.000	30	Mujer


Ello resulta más comprensible que saber que el grupo 1 está formado por los clientes (23, 24, 54, 67, 89, 100, 123, 145, 165). El hecho de tener una lista de sus descripciones no nos aclara mucho las cosas.

Otra forma de sintetizar las características de cada grupo es mediante reglas de este tipo:

$$(Atributo_1 = valor_1) \wedge (Atributo_2 = valor_2) \wedge \dots \wedge (Atributo_m = valor_m) \Rightarrow \text{Grupo 1}$$

Por ejemplo,

$$\begin{aligned} &(Centro = 1) \wedge (Horario = \text{'Mañana'}) \wedge (Act\ 1 = \text{'Yoga'}) \wedge (Act\ 1 = \text{'Stretch'}) \wedge \\ &\wedge (Prof = \text{'Jubilado'}) \wedge (Edad = 68) \wedge (Renta = 3.000.000) \wedge (Sexo = \text{'Mujer'}) \Rightarrow \\ &\Rightarrow \text{Grupo 1} \end{aligned}$$

En este punto ya vemos que pueden surgir algunos problemas. En efecto, la descripción intensiva con valores “medios” parece fácil de obtener a partir de valores numéricos tras haber obtenido la media. Sin embargo, ¿cómo obtenemos la “media” de valores no numéricos como, por ejemplo, los atributos *Horario* o *Prof*? Volveremos a esta cuestión más adelante. 

Podemos definir el **problema de la agregación** en los términos siguientes: dado un conjunto de atributos  $A_1, \dots, A_m$  que toman valores en dominios  $D_1, \dots, D_m$ , un conjunto de  $n$  objetos  $O_1, \dots, O_n$  –cada uno de los mismos descrito como una tupla sobre los atributos  $A_1, \dots, A_m$ –, y un criterio de similitud entre objetos  $S(O_i, O_j)$ , la agregación consiste en encontrar una partición del conjunto de objetos originales en grupos  $G_1, \dots, G_g$ , de manera que todo objeto  $O_i$   $1 < i < n$  pertenezca a un grupo y que, para cada grupo  $G_i$ , todo objeto que pertenezca al mismo tenga una similitud  $S_k$  con los objetos de su grupo que sea mayor que la que éstos exhiben con otra  $G_k$ , con  $k \neq j$ .

### 3. Espacio, distancia y similitud

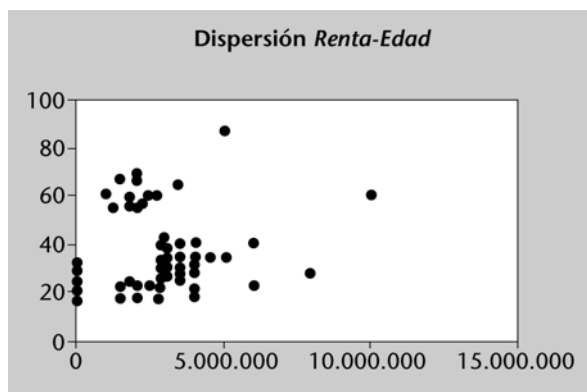
Podemos imaginarnos que los diferentes atributos que describen los objetos que queremos clasificar son los ejes de un espacio.

Los atributos definen los ejes del espacio.

Tomemos un ejemplo sencillo con sólo dos atributos: la renta de un cliente y su edad, que nos define un espacio de dos dimensiones representable mediante dos ejes: el eje  $y$  podemos asignarlo a la edad, y el eje  $x$ , al nivel de renta. Si exploramos el fichero de clientes de Hyper-Gym y observamos los valores que poseen estos dos atributos para cada registro de cliente, podemos representar cada registro y cada cliente en un espacio de dos dimensiones. Por ejemplo, un cliente de treinta y cinco años con una renta de 5,5 millones queda representado en este espacio de dos dimensiones por el punto (35, 5,5); uno de cuarenta años y 3,5 millones, por el punto (40, 3,5); uno de veinte años y 1,2 millones, por (20, 1,2), etc.).

En la figura siguiente podemos observar los diferentes puntos resultantes de proyectar sobre este espacio bidimensional los valores correspondientes a una serie de clientes de Hyper-Gym.

Cada objeto es un punto en el espacio de clasificación.



Aparecen varias nubes de puntos que corresponden a posibles grupos de clientes: los clientes jóvenes con una renta baja, los clientes jóvenes con una renta alta y los clientes de edad media con rentas altas y medias. También, podemos ver una serie de puntos que corresponden a otras combinaciones de valores; incluso hay algunos puntos aislados, sin vecinos próximos, que corresponden a individuos peculiares.

Es importante darse cuenta de que en este espacio podemos definir una **distancia entre objetos** en función de los valores de los atributos correspondientes, que nos permiten asimilar en este espacio los conceptos de *objetos parecidos* y *objetos próximos*.

La distancia nos da un criterio de similitud.

### Ejemplo de distancia entre objetos

Dados dos clientes, representados por dos puntos en el espacio, podemos definir una distancia muy sencilla: por cada atributo (edad o renta) que tengan igual sumaremos un 1 y por cada atributo que sea diferente contaremos un 0. Dividiremos el resultado por el número de atributos que se consideran (dos, en este caso). De esta manera, los objetos idénticos tendrán valor 1 y los objetos completamente diferentes, valor 0.

Evidentemente, esta medida presenta una serie de inconvenientes que ahora no comentaremos, pero nos ofrece una idea de cómo podemos aplicar un primer criterio de similitud: dos objetos son parecidos si su distancia en el espacio de atributos es mínima.

Este tipo de análisis se puede extender a problemas en los que el número de atributos sea mucho mayor que dos. Si disponemos de un conjunto de  $n$  atributos, habrá que hacer la proyección sobre un espacio  $n$ -dimensional. Este hecho limita las posibilidades de extraer alguna conclusión a partir de los datos mediante simple inspección visual. En efecto, podemos representar figuras en dos y en tres dimensiones, pero aquí finalizan nuestras posibilidades de percepción.

Un truco para superar esta dificultad consiste en contemplar subespacios del espacio considerado; por ejemplo, podemos centrarnos en el subespacio bidimensional definido por los atributos *Edad y Renta* o *Edad y Horario*, etc.


El concepto de **distancia entre objetos** nos permite algo más que definir objetos parecidos: nos permite establecer el concepto de *vecino*.

Un **vecino** de un objeto determinado es aquel objeto que se encuentra a una distancia muy próxima en el espacio de clasificación.

¿Qué quiere decir *distancia próxima*? Que el valor de la distancia es inferior a un umbral que se ha establecido como razonable (*a priori* o de manera dinámica). Por ejemplo, si adoptamos la distancia euclidiana entre dos objetos, podemos decir que todo objeto que se encuentre a una distancia inferior a 0,3 de un objeto determinado es su vecino.

Como hemos comentado, el concepto que nos permite agrupar objetos es el de la semejanza o similitud que hay entre éstos.

La **similitud** recoge la intuición de que hay cierta proximidad natural entre los objetos del mismo grupo.

Ahora bien, dado que necesitamos automatizar el proceso de agregación de objetos parecidos, debemos formalizar el concepto de manera que pueda medirlo un ordenador. 

Ahora pasamos a revisar diferentes maneras de definir y medir la similitud.

### 3.1. Medidas de similitud

Como hemos visto, para formalizar el concepto de similitud es útil visualizar el problema dentro de un espacio con tantas dimensiones como atributos haya y asimilar el concepto de *similitud* al de *proximidad entre puntos* de este espacio.

Las **medidas de similitud** intentan establecer cómo se calcula la distancia entre dos puntos en un espacio de  $n$ -dimensiones.

Recordamos que cada uno de estos puntos es la representación en ese espacio abstracto de un objeto del dominio en razón de sus atributos (por ejemplo, un cliente en un espacio bidimensional definido por los atributos *Edad* y *Renta*).

Se trata, pues, de representar cada observación como un punto en el espacio y de medir el grado de proximidad entre dichos puntos. Ahora bien, las cosas no son tan fáciles. De entrada, los atributos que forman las observaciones no siempre tienen las características necesarias para medir fácilmente la similitud.

Recordemos con qué formas aparecen los valores de cada atributo:

1) **Variables numéricas.** Son las variables que toman valores en un conjunto de números ordenado como pueden ser los enteros, los naturales o los reales; por ejemplo, la edad o la temperatura. Dentro de estas variables podemos distinguir los valores que corresponden a medidas auténticas; es decir, los valores que se toman en relación con un valor de origen, como por ejemplo el peso, la longitud y el volumen. La propiedad de este tipo de atributos es que, además de que tiene sentido hablar de orden entre valores, también mantienen propiedades como la proporción. Si alguien tiene cuarenta años, la propiedad *Edad* es el doble que la de alguien que tiene veinte.

2) **Variables categóricas.** Son las variables cuyos valores forman un conjunto sin ninguna relación de orden. Por ejemplo, el tipo de actividad que desarrolla un cliente de Hyper-Gym es una variable de este tipo. Puede hacer yoga, *stretch*, aeróbic o cualquiera de las actividades que hay para elegir. No tiene demasiado sentido decir que 'Yoga' es mayor que 'Stretch'; lo único que sabemos es que la actividad 'Yoga' es diferente de la actividad 'Stretch'.

3) **Rangos.** Las variables que adoptan valores en rangos nos permiten establecer un orden, pero no podemos medir el grado de diferencia que hay entre un valor y otro. Por ejemplo, sabemos que los horarios de mañana son anteriores a los de tarde, pero no sabemos en qué proporción.

**4) Intervalos.** Son conjuntos de valores numéricos; por ejemplo, sabemos que el rango de edad correspondiente a los clientes jóvenes oscila entre dieciocho y treinta y cinco años.

El hecho de que una variable sea de un tipo u otro puede acarrear problemas a la hora de establecer su semejanza en el sentido de proximidad en el espacio. En efecto, las medidas tradicionales de semejanza funcionan correctamente con valores numéricos y variables que toman valores en intervalos muy definidos, pero no se pueden aplicar directamente a variables categóricas; lo más normal es establecer un procedimiento de transformación de los valores de un tipo a otro.


#### Ejemplo de problema con las variables categóricas

La categoría profesional 'Funcionario', por ejemplo, puede tener asignado el valor 0; la categoría 'Profesional autónomo', el valor 1; 'Jubilado', el valor 2, etc. Sin embargo, es necesario tener en cuenta que carece de sentido establecer comparaciones en torno a estos valores. No hay que inferir, por ejemplo, que un jubilado es más parecido a un profesional autónomo que a un funcionario porque la distancia (diferencia) entre sus codificaciones numéricas es menor.

### 3.1.1. Definición y propiedades de las distancias

Hay muchas formas de definir las distancias entre diferentes puntos de un espacio  $n$ -dimensional. En cualquier caso, se ajustan a la definición que establecemos a continuación.

Una **distancia** es una función que toma valores en un espacio  $n$ -dimensional y que da como resultado un valor real positivo. Por norma general, el rango de valores reales positivos se limita al intervalo  $[0, 1]$ .

Asimismo, dados dos puntos  $X_1, X_2$  en un espacio  $n$ -dimensional, para que una función sea calificada como distancia, es preciso que cumpla las propiedades siguientes: 

- $Dist(X_1, X_2) = 0$  si, y sólo si,  $X_1 = X_2$
- $Dist(X_1, X_2) \geq 0$  para todo  $X_1, X_2$
- $Dist(X_1, X_2) = Dist(X_2, X_1)$
- $Dist(X_1, X_2) \leq Dist(X_1, X_3) + Dist(X_3, X_2)$

En términos de semejanza, cada propiedad nos indica lo siguiente:

- El objeto más parecido a un objeto dado es él mismo.
- El valor de la similitud es mayor o igual a 0 para todo par de objetos.
- La similitud es conmutativa.
- Pasar por un punto intermedio  $X_3$  en el camino de  $X_1$  a  $X_2$  no disminuye la distancia total.

## Funciones de distancia más típicas

Recordemos que a la hora de interpretar las diferentes distancias que hemos mencionado, cada objeto es el conjunto de valores correspondientes a cada uno de sus atributos; es decir, es un punto en el espacio que representaremos como un vector.

Si tenemos  $m$  atributos para todas las observaciones, la observación  $i$ ,  $O_i$  queda representada por el vector asociado siguiente:

$$O_i = \langle O_{i1}, \dots, O_{im} \rangle$$

A continuación, mencionamos las distancias más habituales.

### 1) Distancias para valores numéricos

a) Valor absoluto de la diferencia:

$$D_{va}(O_i, O_j) = \sum_{k=1}^m |O_{ik} - O_{jk}|$$

b) Cuadrado de la diferencia:

$$D_{qd}(O_i, O_j) = \sum_{k=1}^m (O_{ik} - O_{jk})^2$$

c) Valor absoluto normalizado:

$$D_{van}(O_i, O_j) = \sum_{k=1}^m |O_{ik} - O_{jk}| / (\text{Diferencia máxima entre los valores de los atributos})$$

d) Distancia euclidiana:

$$D_{eu}(O_i, O_j) = \sqrt{\sum_{k=1}^m |O_{ik} - O_{jk}|}$$

e) Distancia euclidiana normalizada:

$$D_{eu}(O_i, O_j) = \sqrt{\sum_{k=1}^m |O_{ik} - O_{jk}|} / (\text{Diferencia máxima entre los valores de los atributos})$$

Fijaos en que, por lo general, será más interesante trabajar con distancias normalizadas, puesto que nos aseguran que su valor se encuentra entre 0 y 1.



### Ejemplos de distancias entre observaciones

Supongamos que tenemos la serie de clientes que presentamos en la tabla que vemos a continuación:

Cliente	Renta	Años en el club
21	5.000.000	4
212	10.000.000	1
1	6.000.000	2
113	4.000.000	4
221	3.500.000	3
1.234	1.200.000	4
13	5.000.000	2

En este ejemplo basado en dos atributos se hace patente un problema muy real. Las rentas oscilan en un rango de valores entre 1.200.000 y 10.000.000, mientras que los años lo hacen entre 1 y 4. Hay que situar ambos atributos en la misma escala para poder efectuar comparaciones significativas que no den más peso a la renta que a los años de permanencia en el club.

Si calculamos las distancias del cliente número 21 con respecto a los otros mediante las diferentes métricas, tenemos que:

Cliente	$D_{va}$	$D_{qd}$	$D_{van}$	$D_{eu}$	$D_{eun}$
212	5.000.003	$2,5E + 13$	1,00	2.236,07	1,00
1	1.000.002	$1E + 12$	0,80	2.000,00	0,89
113	1.000.000	$4E + 12$	0,40	1.414,21	0,63
221	1.500.001	$2,25E + 12$	0,10	707,11	0,32
1.234	3.800.001	$3,24E + 12$	0,46	1.516,58	0,68
13	2	4	0,76	1.949,36	0,87

Analicemos qué nos indican estos datos. Observamos, por ejemplo, cuáles son los clientes que se parecen más al cliente 21, cuáles son los más próximos, sus vecinos, según cada una de las distancias:

$D_{va}$	$D_{qd}$	$D_{van}$	$D_{eu}$	$D_{eun}$
5.000.003	$2,5E + 13$	1,00	2.236,07	1,00
4.000.001	$1E + 12$	0,80	2.000,00	0,89
2.000.002	$4E + 12$	0,40	1.414,21	0,63
500.001	$2,25E + 12$	0,10	707,11	0,32
2.300.001	$3,24E + 12$	0,46	1.516,58	0,68
3.800.002	$2,25E + 12$	0,76	1.949,36	0,87

Podemos ver que todas las distancias mantienen el mismo orden, salvo la distancia del cuadrado de la diferencia, que da el orden 221, 113, 1.234, 13, 1 y 212. Por lo tanto, cabe esperar que el mismo procedimiento de agregación genere particiones diferentes con distancias diferentes.

En general, la semejanza entre objetos se organiza en forma de **matriz de similitud**, en la cual se miden las distancias de cada objeto con respecto a todos los otros.

### Ejemplo de matriz de similitud

Ésta es la matriz de similitud para los objetos del ejemplo anterior calculada con la distancia euclidiana normalizada:

	21	212	1	113	221	1234	13
21	0,00	1,00	0,89	0,63	0,32	0,68	0,87
212	1,00	0,00					
1	0,89		0,00				
113	0,63			0,00			
221	0,32				0,00		
1.234	0,68					0,00	
13	0,87						0,00

Se trata de una matriz claramente simétrica en la que todos los elementos de la diagonal también son ceros, tal como cabía esperar de las propiedades de las distancias.

Podéis ver al respecto el “Ejemplo de distancias entre observaciones” que presentamos más arriba en este mismo subapartado.

## 2) Distancias para valores categóricos

¿Qué ocurre cuando los valores de los atributos de los objetos cuya proximidad queremos observar son categóricos? No podemos aplicar algunas operaciones matemáticas típicas como la resta o la suma, de manera que nos vemos obligados a recurrir a otros esquemas de comparación.

La **distancia de Hamming** es la operación más sencilla. Esta distancia cuenta el número de atributos diferentes entre dos objetos. Se puede formalizar de la manera siguiente:

$$D_{ham}(O_i, O_j) = \sum_{k=1}^m \delta_{ik}$$

También está la **distancia de Hamming normalizada**:

$$D_{hm}(O_i, O_j) = \sum_{k=1}^m \delta_{ik} / m$$

donde  $\delta_{ik}$  se define como 0 si  $O_{ik} = O_{ij}$  y como 1, en caso contrario.


### Ejemplo de distancia de Hamming

Extraemos los datos del cliente 1 y del cliente 113 y los colocamos en la tabla siguiente. La distancia de Hamming entre ambos es:  $D_{ham}(O_1, O_{113}) = 0 + 0 = 0$ .

Cliente	Sexo	Horario
21	Mujer	Mañana
212	Hombre	Mañana
1	Mujer	Tarde
113	Mujer	Tarde
221	Mujer	Mañana
1.234	Hombre	Tarde
13	Hombre	Tarde

Vemos que la distancia es 0; lo cual se debe a que estos dos atributos son completamente iguales. En cambio, entre el cliente 221 y el 1.234 la distancia es 2, puesto que ambos atributos son completamente diferentes (la distancia normalizada daría 1).

### 3) Combinación de distancias

En las bases de datos que utilizaremos normalmente en el mundo real se mezclan atributos de todo tipo, tanto numéricos como categóricos, de manera que habrá que combinar varias de las funciones que hemos visto en una sola. 


#### Ejemplo de combinación de distancias

Podemos utilizar la distancia euclidiana normalizada para los valores numéricos y sumarla a la de Hamming normalizada para los categóricos. Entonces, la medida de similitud es la siguiente:

$$\begin{aligned} D_{sim}(O_i, O_j) &= D_{eu}(O_i, O_j) = \\ &= \sqrt{\sum_{k=1}^m |O_{ik} - O_{jk}| / (\text{Diferencia máxima entre los valores de los atributos}) + D_{hm}(O_i, O_j)} = \\ &= \sum_{k=1}^m \delta_{ik} / m / 2 \end{aligned}$$

#### Actividad

3.1. Buscad los vecinos más cercanos al cliente 1 con la distancia que hemos definido en el ejemplo anterior. Reflexionad sobre los resultados obtenidos.

Todas las distancias se pueden modificar con el fin de dar más peso a algún atributo o tipo de atributo. Ésta es una manera de indicar que sabemos que algunos atributos poseen más repercusión que otros a la hora de establecer la semejanza entre objetos. Este mecanismo se puede utilizar para codificar algún tipo de **conocimiento a priori** que podamos tener del dominio. 

#### Ejemplo de asignación de pesos

En el caso de la medida de similitud combinada que hemos establecido en el "Ejemplo de combinación de distancias" podemos dar más peso a los atributos categóricos.

## 4. Métodos de agregación

La metáfora espacial y el uso de las medidas de distancia como criterio de similitud nos permiten disponer de una primera herramienta para agregar objetos parecidos. Los métodos de agregación se proponen obtener una enumeración de grupos de objetos parecidos a partir de  $n$  conjuntos de datos. Estos grupos o *clusters*\* ofrecen una primera idea de la estructura del dominio, de cómo están organizados los objetos en razón de sus características. A continuación, estudiamos algunos métodos de agregación de objetos que utilizan criterios parecidos para construir finalmente la colección de grupos de objetos.

\* Del inglés, *agregación, apilamientos*.

### 4.1. El método de los centroides: *k-means*

El **método de los centroides** se basa en la idea de obtener un número  $k$  de grupos que queda fijado al principio del proceso.

#### *K-means*

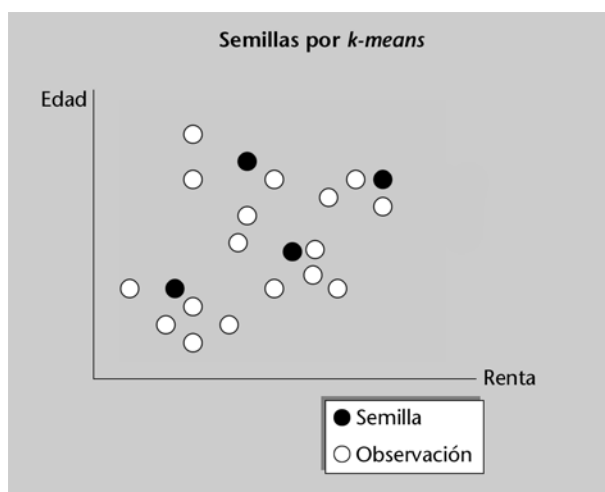
En este método se fija *a priori* el número de grupos que se desea obtener.

Cada uno de estos grupos finales posibles se representa inicialmente con una semilla\*, es decir, se fija un punto inicial del espacio como centro de un grupo potencial. Esta semilla puede ser tanto un objeto seleccionado detalladamente entre el conjunto de objetos inicial como una combinación de valores creada de manera artificial y que corresponda al resumen de las características de varios objetos.

\*En inglés, *seed*.

La idea es muy parecida a la que hemos visto al hablar de la discretización por el método de *k-means*. La diferencia es que en aquel caso los puntos eran los valores de una única variable y las semillas formaban el punto medio –es decir, la media de valores– de los valores que se encontraban en el mismo intervalo. En aquel caso trabajábamos con un espacio unidimensional. !

Podéis ver la discretización por el método de *k-means* en el módulo "Preparación de datos" de esta asignatura.



#### El método de *k-means*

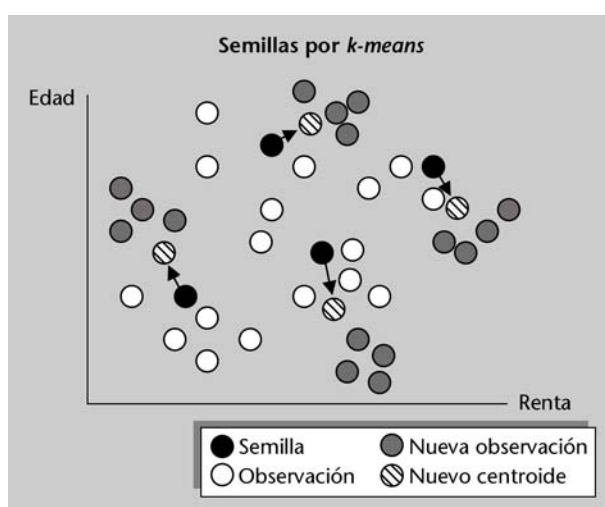
El método de *k-means* consiste en los pasos siguientes:

- 1) Seleccionar las semillas iniciales.
- 2) Calcular los centros.
- 3) Asignar objetos al grupo con centro más próximo.
- 4) Recalcular los centros.
- 5) Continuar hasta que no haya variación en los grupos.

En la figura anterior podemos ver una asignación inicial de centroides para cuatro grupos en un dominio definido sobre dos dimensiones.

A partir de este momento ya podemos comparar los diferentes objetos existentes en el dominio. Para cada objeto, hay que ver qué centroide tiene más cerca. Entonces, se asigna ese objeto al grupo representado por el centroide más próximo.

Como resultado de la agregación de objetos y de considerar todos los objetos próximos entre sí, hay que volver a calcular el centro. Ello se hace obteniendo para cada atributo (o dimensión del espacio) el valor medio de todos los objetos que hay en el grupo. De este manera se desplaza el centro del grupo, paso que se da cuando todos los objetos del dominio han sido asignados a un grupo u otro. Este proceso se esquematiza en la figura siguiente:



Una vez calculados los nuevos centros de los grupos, debemos iniciar otra vez el proceso. Volvemos a considerar la distancia de cada objeto a los diferentes centros, lo asignamos al grupo del centro más próximo y volvemos a calcular centros. ¿Cuándo se acaba el proceso? Cuando entre dos iteraciones seguidas no se produzcan cambios (o no se produzcan demasiados cambios) en la situación de los centros; dicho de otro modo, cuando no se produzcan cambios en los límites de los grupos.

El uso de este método no suele limitarse a esperar el resultado de una iteración. Hay que probar otros valores de  $k$  para saber si la agregación obtenida es mejor o peor, lo cual nos lleva al problema de la calidad del modelo resultante y de la lista de grupos obtenidos.

#### Observación

En cada paso de iteración del proceso de *k-means* puede haber observaciones “tránsfugas” que pasen de un grupo a otro a causa del cambio de centroides.

Podéis ver el problema de la calidad del modelo resultante en el subapartado 5.2 de este módulo didáctico.

## 4.2. El método de los vecinos más cercanos (*k-nearest neighbours*). Proximidad entre grupos

La **proximidad entre grupos** se detecta a partir de la semejanza entre objetos. Podemos calcular esta semejanza a partir de cualquiera de las distancias que ya conocemos.

Podéis ver las distancias más típicas en el subapartado 3.1.1 de este módulo didáctico.

Se define una **matriz de similitud** que conserva para cada objeto  $O_i$  del dominio su distancia con cualquier otro objeto del dominio  $O_j$ . En cada posición  $i, j$  de la matriz se almacena el valor de la distancia entre los objetos  $O_i$  y  $O_j$ . Así, si el dominio consta de  $m$  objetos, la matriz que hay que construir debe ser de  $m \times m$  posiciones. De hecho, de  $(m \times m)/2$  posiciones, puesto que la distancia entre dos objetos es una medida simétrica y basta con almacenar la mitad de los valores, lo que nos proporciona una matriz triangular.

El **procedimiento de los vecinos más cercanos\*** es ahora relativamente sencillo:

\* En inglés, *nearest neighbours*.

- 1) Encontrar el valor mínimo de la matriz de similitud. Indica los dos grupos (inicialmente objetos solos) que están más cerca.
- 2) Fusionar los dos grupos. Implica eliminar la fila correspondiente a uno de los dos grupos (el que ha sido absorbido).
- 3) Recalcular los valores de la matriz calculando la distancia del resto de los grupos al nuevo grupo.

Un aspecto importante para decidir y poder efectuar estas operaciones es cómo podemos calcular la distancia entre varias agregaciones. !

En el primer paso, la distancia entre grupos no es diferente a la distancia entre objetos. De hecho, todos los grupos son grupos de un solo objeto. Ahora bien, cuando iniciamos el proceso de agregación, debemos tomar una decisión respecto a cuál es el conjunto de valores para cada atributo que representa los objetos del dominio y que, por tanto, utilizamos para medir la distancia con respecto a los otros grupos. Básicamente hay tres criterios.

- 1) **Comparar centroides**: compara la distancia entre los centros de cada grupo. Los centros se calculan como la media de valores de todos los objetos para cada atributo.
- 2) **Comparación de enlace sencillo**: la distancia entre dos grupos es la distancia entre los objetos que están más cerca.
- 3) **Comparación de enlace completo**: la distancia entre grupos es la distancia entre los objetos más distantes.

Acabamos de establecer la base del método conocido como método de los  $k$ -vecinos más cercanos. !

#### 4.3. Métodos incrementales o aglomeradores

En el método de  $k$ -means siempre se empieza por un número fijo de grupos conocidos *a priori*, lo cual lo hace útil cuando tenemos idea de cuántos grupos hay en realidad. Por ejemplo, cuando sabemos que hay tres o cuatro tipos de clientes y queremos conocer sus características, utilizamos como semilla al cliente más ca-

##### Aglomeración

En los métodos aglomeradores no hay ningún número de grupos fijado *a priori*.

racterístico de cada grupo. Pero hay situaciones en las que nuestro desconocimiento del dominio de aplicación es todavía mayor. Por lo tanto, ni siquiera es posible fijar una cantidad  $k$  de puntos iniciales alrededor de los cuales podamos ir formando los grupos. En este caso, hay que reflejar este desconocimiento adoptando una actitud neutra con respecto a los datos. Una de las maneras de resolver el problema es mediante los *métodos aglomeradores*.

Los **métodos aglomeradores** empiezan considerando que cada objeto forma un grupo por sí mismo (un grupo de un solo elemento) y a partir de ahí, evalúan las distancias entre grupos (u objetos, en el primer paso), y crean los diferentes grupos finales por aglomeración.

En el proceso se detectan situaciones en las que, por ejemplo, los objetos de dos grupos están tan cerca, resultan tan parecidos, que hay que fusionarlos en uno sólo. También podemos encontrarnos con la situación contraria, en cuyo caso hay que dividir un grupo en dos. En las primeras fases del proceso, los grupos son muy pequeños y muy diferentes entre sí, y además están bien definidos. En las últimas fases, los grupos son mayores.

Gráficamente, se puede interpretar como si el proceso crease un “árbol”,\* en el que la raíz representa todo el conjunto de observaciones, y las hojas, cada observación concreta. El proceso puede expresarse con el siguiente esquema algorítmico:

\* ¡Pero este “árbol” no tiene nada que ver con los árboles de decisión!

- 1) Crear un árbol con un nodo único que represente todo el conjunto de observaciones.
- 2) Mientras no se cumpla el criterio de finalización, hay que ejecutar el bucle siguiente:

**Para cada** nueva observación  $O_i$  **hacer**  
**Para cada** nodo  $n_j$  **hacer**  
    Si la calidad de  $n_j$  mejora al incorporar  $O_i$ , **entonces**  $n_j = n_j \cup O_i$   
    Si no se encuentra ningún nodo  $n_j$  que mejore,  
    **entonces** se crea uno nuevo nodo únicamente con  $O_i$   
**fpara**  
**fpara**

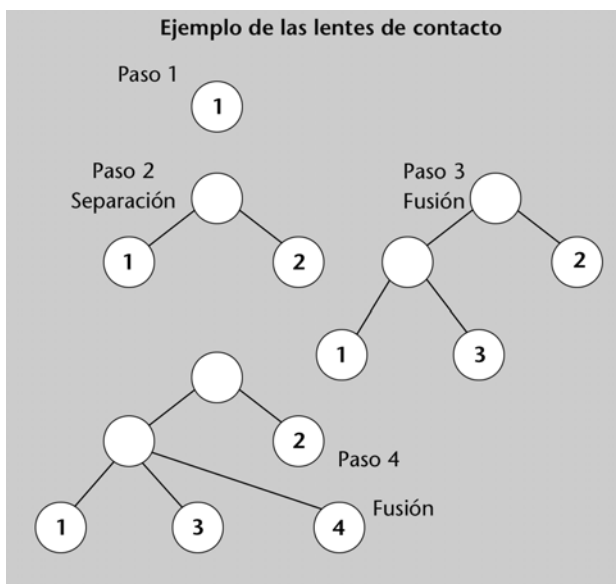
- 3) Anotar cuál es el nodo con mejor calidad  $n_{max}$  y el siguiente en calidad  $n_{max-1}$ . Entonces:

a) Si al unir  $n_{max}$  y  $n_{max-1}$  da un nodo con más calidad que  $Max(calidad(n_{max}), calidad(n_{max-1}))$ , entonces se crea un nuevo nodo que es resultado de la fusión de  $n_{max}$  y  $n_{max-1}$ .

b) En caso contrario, es necesario considerar si es mejor dividir  $n_{max}$  en dos nodos, comparando su calidad con la de las particiones resultantes.

Gráficamente, podemos ver el comienzo con el ejemplo de las lentes de contacto, en el que dejamos de considerar el atributo de recomendación como etiqueta de clase. Utilizamos la distancia de Hamming normalizada.

Podéis ver el "Ejemplo de las lentes de contacto" en el subapartado 2.1. del módulo "Clasificación: árboles de decisión" de la presente asignatura.



- 1) El primer elemento del conjunto de datos es {'Joven', 'Miope', 'Sí', 'Normal', 'Duras'}; crea un *cluster* por sí mismo.
- 2) El segundo elemento es {'Joven', 'Hipermetrope', 'Sí', 'Normal', 'Duras'}; su distancia es de 1/5. Consideramos que puede crear su propio *cluster*. Ya tenemos la primera estructura arborescente.
- 3) El tercer elemento es {'Prepresbicia', 'Miope', 'Sí', 'Normal', 'Duras'}, y tiene las distancias siguientes: 1/5, al primer elemento, y 2/5, al segundo. Consideramos que es mejor agrupar el primer y tercer elementos.
- 4) El cuarto elemento es {'Presbicia', 'Miope', 'Sí', 'Normal', 'Duras'}. Las distancias son: 1/5 al primer elemento, 2/5 al segundo y 1/5 al tercero. Por lo tanto, lo integramos en el *cluster* (1,3).

Debemos precisar algunos aspectos más para poder detallar el funcionamiento de este método.

En primer lugar, lo que hemos dado como algoritmo es en realidad un esquema que puede convertirse en varios algoritmos variando los criterios siguientes: la condición de final, la detección de la proximidad entre grupos y el cálculo de la calidad de cada grupo o nodo. !



En segundo lugar, hacemos hincapié en un aspecto importante: explicaremos primero los dos pasos que se efectúan una vez hemos visto que la nueva observación se puede añadir a uno de los grupos existentes o bien veremos si es mejor que forme su propio grupo.

1) Como podemos ver, se hace una comparación entre la calidad de los dos grupos mejores y, dependiendo del resultado, se fusionan. La idea es que la única manera de cambiar la estructura del *clustering* no sea mediante la aportación de instancias adicionales. En efecto, si continuáramos así el resultado final dependería en gran medida del orden de llegada de las observaciones. Éste es uno de los problemas graves que presentan los métodos incrementales y hay que definir métodos para poder compensarlos.

2) Una vez se ha considerado la nueva observación, o bien antes, es necesario evaluar la calidad global de la partición en grupos existentes en un momento determinado. Una manera de hacerlo sería considerar si vale la pena fusionar algunos de los grupos actuales. Por supuesto, dicha posibilidad introduce una ineficiencia grave, ya que para  $n$  clusters tenemos

$$\binom{n}{2} = \frac{n(n-1)}{2}$$


combinaciones posibles de dos clusters que nos permiten comprobar la calidad de la partición resultante.

Por este motivo, en general, se tiende a la solución de compromiso que representa guardar una memoria de los dos mejores clusters; también se efectúan las comprobaciones que hemos recogido en el último paso (la fusión de los nodos) del esquema algorítmico que hemos presentado.

### Criterio de parada

El proceso continúa hasta que se cumpla algún criterio de parada válido. Por norma general, ese criterio consiste en que se mantenga una distancia determinada entre grupos. También se puede continuar el proceso hasta que sólo haya un grupo. Como en cada paso queda registrada la fusión que se ha producido, obtenemos la historia del proceso y podemos ver en qué nivel interesa detenerse.

### Detección de la proximidad entre grupos

El problema que se puede producir es que se generen particiones demasiado atomizadas, en el peor de los casos con una gran preponderancia de las particiones con un único objeto, o, lo que es lo mismo, que se produzcan problemas de sobreespecialización\*. 

No obstante, por lo general, se da un **parámetro de corte** que detecta e impide un crecimiento excesivo del árbol de agregación. La idea es que si se trata de

#### Lecturas complementarias

Encontraréis métodos que compensan los problemas de los métodos incrementales en las obras siguientes:

**J. Béjar** (1995). *Adquisición de conocimiento en dominios poco estructurados*. Tesis doctoral. Barcelona: Universidad Politécnica de Cataluña, Departamento de Lenguajes y Sistemas Informáticos.

**D. Fisher** (1987). "Knowledge Acquisition via Incremental Conceptual Clustering". *Machine Learning* (vol. 2, núm. 2, pág. 139-172).

**J. Roble**, (1998) "Robust Incremental Clustering with Bad Instance Orderings: a New Strategy". En: H. Coelho (ed.). *Progress in Artificial Intelligence: IBERAMIA '98. Sixth Iberoamerican Conference on Artificial Intelligence* (pág. 136-147). Lisboa: Springer Verlag.

\* En inglés, *overfitting*.

una observación lo suficientemente parecida a las existentes en la actualidad, se puede suponer que no aporta ninguna información significativa; de manera que no deberemos considerarla. Cuando una observación nueva no incrementa en la medida suficiente la calidad de ningún grupo, es decir, cuando la calidad nueva no es más alta que la anterior más el valor mínimo indicado por el parámetro de corte, simplemente no la tenemos en cuenta.

#### Agregación jerárquica

La agregación jerárquica permite varios niveles de concreción en la descripción de los grupos existentes.

### Calidad de una agregación

En principio, la calidad de una agregación se puede medir de varias maneras; pero, en cualquier caso, se trata de asegurar que una clase recibe un valor de calidad más alto cuanto más alta sea la similitud entre las observaciones que reúne y cuanto más baja sea con respecto a las de las otras clases. En el subapartado siguiente ponemos un ejemplo concreto.

## 4.4. Métodos de agregación probabilistas

### El método COBWEB


Una variación interesante del método anterior es la que aporta el método COBWEB. Hasta ahora hemos visto una serie de métodos de agregación que tomaban elementos dentro del espacio de observaciones según su similitud como criterios para aproximar. Asimismo, dado que trabajábamos sin ninguna guía previa de cuáles o cuántas observaciones había (los criterios para detener la construcción del modelo), la división de grupos existentes venía dada por una serie de medidas de calidad de la partición en los grupos correspondientes.

#### Lectura complementaria

Encontraréis el método COBWEB en la obra siguiente:

D. Fisher (1987). "Knowledge Acquisition via Incremental Conceptual Clustering". *Machine Learning* (vol. 2, núm. 2, pág. 139-172).

Desde el punto de vista estadístico, el objetivo de un procedimiento de agregación es obtener la partición más probable entre todas las posibles a partir de un conjunto de observaciones.

El método COBWEB sigue el esquema que acabamos de comentar. Lo más interesante de este método es la manera de calcular la calidad de cada grupo. 

El concepto utilizado por Fisher es la **utilidad de una categoría** ( $U_c$ ). Para una agregación, grupo, *cluster* o categoría (como quiera llamarse) dentro del conjunto de categorías en un momento dado  $C_1, \dots, C_n$  sobre un dominio definido por los atributos  $X_1, \dots, X_m$ , como cada atributo puede adoptar valores en  $x_{i1}, \dots, x_{iq}$ , la utilidad de una categoría se define como:

$$U_c = \frac{1}{n} \left[ \sum_{k=1}^n P(C_k) \sum_i \sum_j \left( P(X_i = x_{ij} | C_k)^2 - \sum_i \sum_j P(X_i = x_{ij})^2 \right) \right]$$

A continuación definimos cada uno de estos elementos:

- $P(C_k)$  es la frecuencia relativa de la agrupación  $P(C_k)$ ; es decir, la proporción de observaciones del conjunto original de datos que pertenecen a la agrupación  $C_k$ .
- $P(X_i = x_{ij})$  es la probabilidad de que el atributo  $X_i$  adquiera el valor  $x_{ij}$ .
- $P(X_i = x_{ij}|C_k)$  es la probabilidad de que el atributo  $X_i$  tome el valor  $x_{ij}$ , dado el hecho de que pertenece a la clase  $C_k$ .

Esta expresión tan terrorífica no debe ocultarnos la simplicidad de la intuición subyacente. En efecto, en esta expresión podemos distinguir dos términos:

a) La **predicción informada (IG)** o probabilidad de que el atributo  $X_i$  tome el valor  $x_{ij}$ , sabiendo que pertenece a la clase  $C_k$ —probabilidad condicional que, a su vez, está matizada por la probabilidad de la clase  $C_k$ , representada por  $P(C_k)$ —. También se puede interpretar como el número de valores de los atributos que es posible predecir correctamente, atendiendo a la partición jerárquica de las agrupaciones existentes en un momento determinado.

IG es la sigla de la expresión inglesa *Informed Guess*.

b) El **término UG** corresponde al número de valores de los atributos que podrían predecirse correctamente si no supiéramos cuál es la estructura actual de los *clusters*.

UG es la sigla de la expresión inglesa *Uninformed Guess*.

Así pues, podemos expresar toda la fórmula anterior de la manera siguiente:

$$U = \frac{IG - UG}{n}$$

donde  $n$  es el número de agregaciones en un momento dado.

El **criterio de preferencia de COBWEB** crea la jerarquía de *clusters* que maximiza el número de valores de los atributos que pueden predecirse correctamente en una observación nueva, una vez tenemos la información con respecto al grupo o la agregación sobre la que puede recaer esta nueva observación.

El método no guarda las descripciones precisas de las agregaciones, sino una representación que indica para cada par (atributo, valor) la probabilidad de que una observación tenga esa combinación de atributo y valor. Cada nodo del árbol de *clustering* posee una cabecera en la que se indica la frecuencia con que una observación se incluye en la categoría. El resto del nodo contiene una tabla que expresa la frecuencia relativa de aparición de cada valor atributo par. Con esta información se calcula la utilidad de la agrupación que permite apli-

car el método aglomerador descrito en el esquema algorítmico anterior. Como resulta útil, se decide subdividir grupos, fusionarlos o crear un nodo nuevo únicamente con la observación más reciente.

### Ejemplo de aplicación del método COBWEB

Supongamos que tenemos la descripción de este conjunto de socios de Hyper-Gym con relación a un grupo reducido de atributos:

Entrenador personal	Horario	Distrito
Sí	Mañana	A
No	Mañana	B
No	Tarde	B
Sí	Tarde	C

Podríamos obtener la estructura de agregaciones siguiente:

$P(N_1) = 4/4$			$P(X/C)$	
Entrenador	Sí	0,5	Mañana	0,5
	No	0,5		0,5
Horario	Mañana	0,5	Tarde	0,5
		0,5		0,5
Distrito	A	0,25	B	0,5
		0,5		0,5
		0,25		0,25

$P(N_2) = 1/4$			$P(X/C)$	
Entrenador	Sí	0,5	Mañana	0,5
	No	0,5		0,5
Horario	Mañana	0,5	Tarde	0,5
		0,5		0,5
Distrito	A	0,25	B	0,5
		0,5		0,5
		0,25		0,25

$P(N_3) = 2/4$			$P(X/C)$	
Entrenador	Sí	0,0	Mañana	1,0
	No	1,0		1,0
Horario	Mañana	1,0	Tarde	0,0
		0,0		0,0
Distrito	A	1,0	B	0,0
		0,0		0,0
		0,0		0,0

$P(N_6) = 1/4$			$P(X/C)$	
Entrenador	Sí	1,0	Mañana	0,0
	No	0,0		0,0
Horario	Mañana	0,0	Tarde	0,0
		0,0		0,0
Distrito	A	0,0	B	0,0
		0,0		0,0
		1,0		0,0

$P(N_4) = 1/2$			$P(X/C)$	
Entrenador	Sí	0,0	Mañana	1,0
	No	1,0		0,0
Horario	Mañana	1,0	Tarde	0,0
		0,0		0,0
Distrito	A	0,0	B	1,0
		0,0		0,0
		0,0		0,0

$P(N_6) = 1/2$			$P(X/C)$	
Entrenador	Sí	0,0	Mañana	0,0
	No	1,0		1,0
Horario	Mañana	0,0	Tarde	1,0
		0,0		0,0
Distrito	A	0,0	B	1,0
		0,0		0,0
		0,0		0,0

### El método Autoclass

A continuación, presentamos otro ejemplo de método probabilista bastante interesante: el método Autoclass. En este método se considera que la pertenencia de cada observación a una clase o a otra no se puede determinar tajantemente (recurriendo a un umbral de similitud, por ejemplo), sino que se puede dar la probabilidad de que la observación  $X$  pertenezca a la clase  $C$ , y esto para cada una de las clases.

Como no sabemos *a priori* qué conjunto de grupos o clases hay, ni mucho menos el número de particiones óptimo, el espacio para explorar (el espacio de todas las particiones posibles) es enorme. !

Sabemos que las observaciones definidas sobre un dominio  $X_1, \dots, X_n$  siguen una distribución de probabilidad  $P(X_1, \dots, X_n)$  que nos da la probabilidad para cada tupla de valores observados. Además, si conociéramos el número de grupos (clases) existentes, podríamos reconstruir la distribución de probabilidad que rige los valores de las observaciones que pertenecen a cada grupo o clase.

Por otra parte, dado el mismo conjunto de datos definido sobre el mismo dominio, la partición posible en clases sigue otra distribución de probabilidad.

Debemos hacer una serie de suposiciones para aprovechar las propiedades de cada una de estas distribuciones de probabilidad, relacionarlas entre sí y crear un criterio operativo para construir a partir de los datos no sólo una partición, sino la partición más probable. !

Supongamos que tenemos un número de clases determinado (pongamos  $k$ ). Los valores de las observaciones que se encuentran dentro de cada clase  $C_i$  siguen una distribución  $P_{C_i}(X_1, \dots, X_n)$  determinada, que no tiene por qué presentar las mismas características que los valores de las observaciones que se encuentran en otro grupo  $C_j$ ,  $P_{C_j}(X_1, \dots, X_n)$ .

En general, los valores de las observaciones de los conjuntos de datos siguen lo que se llama **modelo de mixtura** (mezcla) de distribuciones. Si tenemos  $k$  clases, en conjunto, la distribución corresponderá a un modelo de  $k$ -mixtura.

Es importante subrayar que cada distribución  $P_{C_i}$  da la probabilidad de que una observación  $X_1, \dots, X_n$  tome los valores  $x_1, \dots, x_n$ , suponiendo que pertenezca a la clase  $C_i$ . La paradoja es que sabemos que cada observación pertenece a un grupo y sólo a uno, pero no sabemos exactamente a cuál. Además, no todos los grupos pueden darse con la misma probabilidad. La mixtura nos da la probabilidad de cada grupo.

Simplificaremos las cosas para poder seguir la discusión sin demasiadas dificultades. Vamos a definir el caso reducido del modelo Autoclass. !

Supongamos que tenemos un atributo numérico,  $X_1$ , y sólo uno, a partir del cual queremos encontrar el conjunto de grupos más probables. Supongamos, también, que ese atributo sigue para cada grupo (que existe, pero que todavía no hemos formado) una distribución normal.

### Lectura complementaria

Encontraréis el método Autoclass en la obra siguiente:

**P. Cheeseman; J. Stutz** (1995). "Bayesian Classification (Autoclass): theory and results". En: U.M. Fayyad; G. Piatetsky-Sahpiro; P. Smyth; R. Uthurusamy (ed.). *Advances in Knowledge Discovery and Data Mining* (pág. 153-180). AAAI Press.

### Lectura recomendada

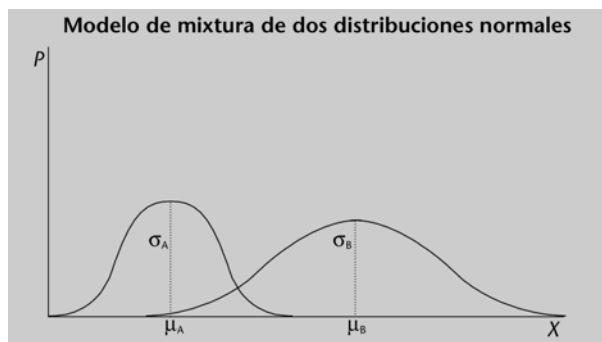
Aquí hemos hecho una reducción de la exposición de Cheeseman para no entrar en las complejidades matemáticas que presenta su método. Para una exposición más general en la que se explica con más detalle el algoritmo Autoclass siguiendo el caso general y no el restringido que ahora utilizaremos, podéis consultar la obra siguiente:

**P. Cheeseman; J. Stutz** (1995). "Bayesian Classification (Autoclass): theory and results". En: U.M. Fayyad; G. Piatetsky-Sahpiro; P. Smyth; R. Uthurusamy (ed.). *Advances in Knowledge Discovery and Data Mining* (pág. 153-180). AAAI Press.

Como cada grupo puede seguir una distribución diferente, la suposición que hacemos es equivalente a decir que cada grupo sigue una distribución normal, pero con parámetros diferentes. Los parámetros que determinan una distribución normal o gaussiana son la media y su varianza. Supongamos que sólo tenemos dos clases; en ese caso, conseguiremos dos distribuciones gaussianas diferentes A y B:

Grupo	Media	Varianza	Probabilidad
A	$\mu_A$	$\sigma_A$	$P_A$
B	$\mu_B$	$\sigma_B$	$P_B$

La mixtura de dos distribuciones normales puede representarse gráficamente de la manera siguiente:



El problema es cuándo sabemos que, en efecto, hay estas dos clases. Entonces, podemos considerar nuestras observaciones como muestras extraídas de la población definida por la mixtura de las dos distribuciones. El problema consiste en determinar las características de las clases a partir de la muestra (de las observaciones utilizadas); es decir, determinar las medias y varianzas respectivas y la probabilidad de una de las dos clases (la otra puede extraerse haciendo  $1 - \text{la probabilidad encontrada}$ ; por ejemplo,  $1 - P_B$ ).


Si supiéramos de cuál de las dos distribuciones procede cada una de las observaciones que forman la muestra, el problema sería muy sencillo; por ejemplo, un estimador correcto de la media de la población es la media de la muestra. Por lo tanto, podríamos extraer la media y la desviación de todas las observaciones que proceden de la clase A, y hacer lo mismo para las que proceden de la clase B. El problema es que no sabemos de qué clase procede cada observación.

Cuando ya conocemos los parámetros que hay que conocer (las medias y las varianzas de la población), la probabilidad de que una observación  $X_1$  proceda de la clase A se puede encontrar mediante la operación siguiente:


$$P(A|X_1) = \frac{P(X_1|A)}{P(X_1)} = \frac{G(X_1, \mu_A, \sigma_A)}{P(X_1)}$$

donde  $G$  es la función de la distribución normal (gausiana) para la clase A. Lo mismo puede afirmarse de la clase B:

$$P(B|X_1) = \frac{P(X_1|B)}{P(X_1)} = \frac{G(X_1, \mu_B, \sigma_B)}{P(X_1)}$$

En consecuencia, si supiésemos que hay dos clases y a qué clase pertenece cada observación, podríamos extraer estas dos distribuciones de probabilidad a partir de la muestra y las distribuciones nos permitirían decir cuál es la probabilidad de que una observación pertenezca a una clase u otra. Aunque, en este caso, no nos hallamos en esa situación. 

#### 4.5. Métodos probabilistas de construcción de agregaciones cuando el número de clases es conocido *a priori*

Supongamos que tenemos la certeza de que hay  $k$  particiones y disponemos de un conjunto de observaciones; es decir, tenemos valores pero desconocemos a cuál de las  $k$  clases pertenece cada valor. ¿Cómo podemos obtener la información que permita asignar observaciones a las clases? El método diseñado en el subapartado anterior nos permite encontrar la probabilidad de que la clase sea  $C_i$  (A o B), a partir de conocer un dato determinado,  $X_1$ . Por lo tanto, para cada observación podemos extraer la probabilidad de que la observación pertenezca a cada clase. Es una manera de expresar las particiones que sabemos que hay. 

La cuestión es que para este reducido problema hay que estimar cinco parámetros: las dos medias, las dos varianzas y una de las probabilidades.

Uno de los métodos utilizados para hacer esta estimación es el método EM. De forma esquemática, el método consiste en ejecutar los pasos siguientes:

EM es la sigla de *Expectation Maximization*, que podemos traducir como 'maximización de la esperanza'.

- 1) Utilizar valores semilla para los cinco parámetros.
- 2) Utilizar los valores semilla para calcular las probabilidades de pertenencia de cada observación a cada clase.
- 3) Estimar otra vez los parámetros con la división en clases que han generado los parámetros estimados de entrada.
- 4) Repetir el proceso.

El criterio para detener el proceso consiste en que entre dos iteraciones sucesivas no haya ningún incremento de la calidad de la partición obtenida. Ahora bien, debemos utilizar alguna medida de calidad.

La **verosimilitud**\* mide hasta qué punto los datos son una muestra que se puede haber extraído de una población que siga las distribuciones estimadas. Es una medida de ajuste que indica hasta qué punto los datos se ajustan a la distribución; por lo tanto, señala hasta qué punto corresponden a la partición obtenida.

\* En inglés, *likelihood*.

Para calcular la verosimilitud, se considera cada una de las observaciones y se determina cuál es su probabilidad. Obtenemos esta probabilidad considerando lo que sabemos de la probabilidad de cada clase, la probabilidad de que la instancia  $X_i$  pertenezca a la clase A y la probabilidad de que pertenezca a la clase B.

$$\prod_i^m (p_A P(X_i | A) + p_B P(X_i | B))$$

A continuación, definimos cada elemento:

- $p_A$  ( $p_B$ ) es la probabilidad de la clase A (B).
- $m$  es el número de observaciones presentes en los datos.

El multiplicatorio se debe a que suponemos que todas las observaciones son independientes entre sí. Recordad que estamos siguiendo la discusión sólo para el caso de un único atributo y dos clases. Normalmente, el valor de la verosimilitud se calcula aplicando logaritmos (es decir, por comodidad y simplicidad se utiliza el logaritmo de la verosimilitud, de manera que todos los productos pasan a ser sumas).

Si en lugar de suponer que hay dos clases suponemos que hay  $k$ , la expresión correspondiente es la siguiente:

$$\prod_i^m \sum_{j=1}^k p_j P(X_i | C_j)$$

Si suponemos  $q$  atributos en lugar de uno sólo, y si suponemos que los atributos son independientes entre sí, tenemos la expresión siguiente:

$$\prod_i^m \prod_{r=1}^q \sum_{j=1}^k p_j P(x_{ir} | C_j)$$

A continuación, añadimos algunos comentarios:


**a)** La suposición de que las observaciones son independientes entre sí es muy fuerte, e incluso puede resultar contraintuitiva. En caso de que haya atributos  $X$ ,  $Y$  de los que se sabe a ciencia cierta que no son independientes, sino que están fuertemente relacionados, no podemos mantener esta suposición. Es preciso modelar su comportamiento para una distribución conjunta  $P(X, Y)$  con su propia media y en la que las varianzas estén ahora representadas por una matriz de covarianza.

**b)** En caso de que se trate de  $X_1, \dots, X_n$  atributos que suponemos dependientes entre sí, entonces debemos modelarlos a partir de una distribución multivariante. La matriz de covarianza posee dimensiones  $n \times n$ . Hay que considerar,



pues,  $n$  medias y  $n(n+1)/2$  elementos de la matriz de covarianza (que es simétrica). El número de parámetros por estimar es, por tanto, cada vez mayor.

c) Finalmente, los atributos categóricos no se pueden modelizar siguiendo una distribución gaussiana.

Los problemas de este tipo de métodos son los que suelen aparecer en todos los procesos de aprendizaje. Podemos llegar a obtener modelos sobre-especializados, bien porque ponemos un número inicial de clases demasiado elevado, o porque damos un número demasiado elevado de parámetros para estimar. 

A continuación, nos centraremos en el problema realmente difícil, que es el que ataca el método Autoclass.

#### 4.6. Métodos de construcción de agregaciones cuando el número de clases es desconocido *a priori*

En el caso anterior suponíamos que partíamos de un conjunto de observaciones y queríamos obtener una agregación con un número fijado de clases. Ahora nos interesa obtener una partición buena sin indicar de entrada el número de clases que queremos obtener. El método Autoclass hace exactamente eso.


El método Autoclass recurre a una técnica llamada *estimación bayesiana*, de la que hablamos en otro módulo. El objetivo del método es obtener la partición en clases que se ajuste mejor a los datos. Ahora, a fin de evitar algunos problemas de sobre-especialización que aparecen usando los métodos anteriores, imponemos una penalización para cada parámetro nuevo introducido. Cada parámetro posee una distribución *a priori*. La introducción de un parámetro nuevo en la verosimilitud origina que tengamos que multiplicar la verosimilitud por la probabilidad *a priori* del parámetro. Como la probabilidad es un valor entre 0 y 1, en principio la reducirá. Por lo tanto, a menos que la calidad general del modelo mejore mucho en combinación con otros parámetros, la introducción de parámetros nuevos penaliza el modelo.

##### Lectura complementaria

Encontraréis con más detalle el método Autoclass en la obra siguiente:

**P. Cheeseman; J. Stutz** (1995). "Bayesian Classification (Autoclass): theory and results". En: U.M. Fayyad; G. Piatetsky-Sahpiro; P. Smyth; R. Uthurusamy (ed.). *Advances in Knowledge Discovery and Data Mining* (pág. 153-180). AAAI Press.

El **método Autoclass** utiliza igualmente el modelo de mixtura de distribuciones (no necesariamente gaussianas) y emplea el método EM para estimar los parámetros de las distribuciones que se ajusten mejor a los datos para diferentes números de clases posibles. Es decir, hay dos niveles de búsqueda: uno para el número de clases y otro para el número de atributos.

Una descripción detallada de este algoritmo supera los ámbitos de la asignatura que ahora nos ocupa, de manera que remitimos al lector a la exposición de Cheeseman –en Fayyad, 1996–, que es la última versión revisada del algoritmo para construir y valorar la calidad de los *clusters* obtenidos por procedimientos probabilísticos. También se pueden seguir otros criterios, como por ejemplo el criterio de información bayesiana o BIC. 

## 5. Interpretación de los modelos obtenidos

Una vez obtenido un modelo de agrupación, debemos investigar qué nos indica. El procedimiento que habrá que seguir va a consistir en la realización de los pasos siguientes:

1) Estudiar cada grupo, lo cual quiere decir que hay que encontrar las características propias de cada grupo. Hay varias maneras de llevar a cabo este estudio:

a) El paso más sencillo consiste en efectuar un análisis estadístico sencillo que observe cuáles son los valores medios de cada atributo.

b) Otra manera es comparar el valor medio de cada atributo dentro de cada grupo con el valor medio en el resto del dominio. Al calcular la diferencia de valores, podemos obtener una idea de cuáles son los atributos más diferentes del grupo; es decir, aquellos que marcan más su diferencia con respecto al resto del dominio.

2) Estudiar la relación entre grupos. Las diferencias entre atributos se pueden precisar más si hacemos las comparaciones grupo a grupo. Establezcamos con qué grupos y variables hay más diferencia e intentamos establecer la razón de esas diferencias.

3) Obtener una descripción más sintética. La descripción que el modelo de agregación da de un dominio es muy limitada y, a veces, poco natural. De hecho, tendremos una enumeración de los objetos de cada grupo o una lista de los valores que tienen los representantes (centros) de cada grupo. Es interesante aplicar a cada grupo algún otro método que nos dé una descripción de nivel más alto, como por ejemplo una colección de reglas que describan cada grupo.

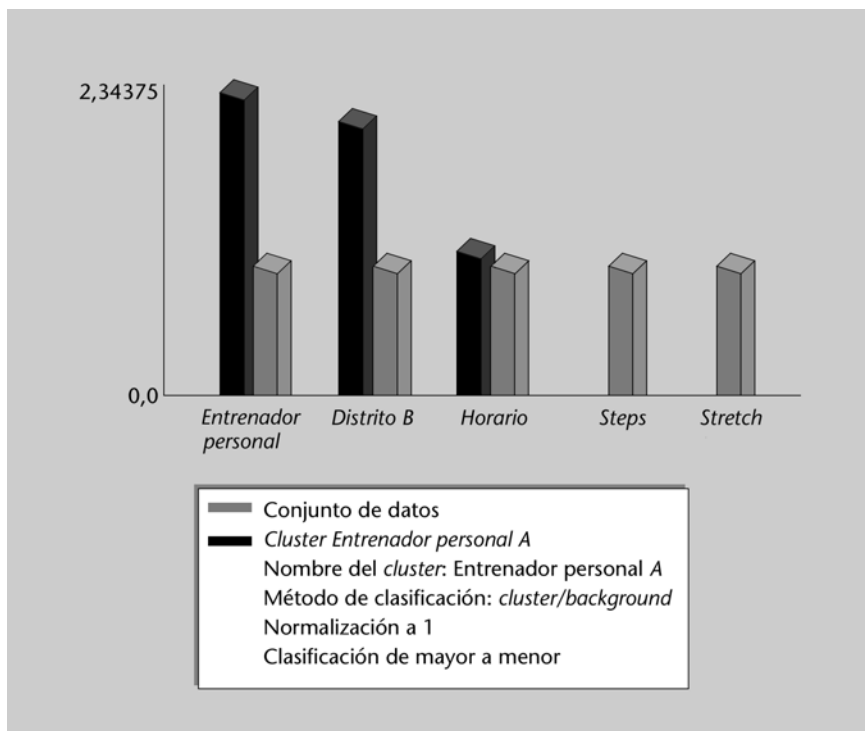
### Ejemplo de interpretación de un modelo de agregación

Utilizando los datos de los ciento cincuenta clientes del gimnasio Hyper-Gym, y con la ayuda de una herramienta como Cviz, que permite efectuar agregaciones y visualizarlas, obtenemos la interpretación que encontraremos en los gráficos de la página siguiente.

Este estudio nos facilita una primera aproximación que nos indica que los *clusters* más importantes están definidos por la conjunción de las variables *Entrenador personal* y *Distrito A*; *Entrenador personal* y *Distrito B*, y *Horario*.



Podemos estudiar la distribución de valores de cada uno de los *clusters*. Por ejemplo, la figura siguiente representa la distribución de valores para el *cluster Entrenador personal y Distrito B*, que indica que los clientes que solicitan entrenador personal y residen en el distrito B hacen más *steps* y *stretch* que el resto con respecto a la media de todas las observaciones.



La tarea de interpretación debe continuar utilizando otros criterios: dispersión de valores por el *cluster* y proximidad a otros *clusters*.

## 5.1. Predicción a partir de los métodos de agregación

La utilidad básica de los métodos de agregación reside en la capacidad de agrupar objetos parecidos. Por ejemplo, queremos hacernos una idea de los grupos de clientes que tenemos. Podemos aplicar un método aglomerador incremental y ver cuántos grupos resultan, o podemos hacer un número de grupos determinado, por ejemplo cinco, y estudiarlos para ver qué obtenemos. Una segunda aplicación es la predicción. Ahora veremos cómo puede llevarse a cabo.

**Predecir** consiste en calcular, a partir de unos valores observados, otro valor que no se ha observado.

Si disponemos de un modelo de agregación, la predicción se basará en el concepto ya conocido de semejanza o proximidad de objetos en el espacio. En efecto, según los atributos observados, podemos establecer qué conjunto de objetos ya conocidos se parecen más al recién llegado e indicar qué valor del atributo desconocido corresponde a los objetos ya conocidos.

Esto plantea algún que otro problema. Por ejemplo, supongamos que tenemos un cliente nuevo en el club, el cliente 9.879, que presenta los datos siguientes:

Cliente	Edad	Sexo	Horario	Renta	Años
9.879	43	Mujer	¿?	5.000.000	¿?

Evidentemente, todavía no sabemos qué horario quiere elegir ni cuánto tiempo es probable que pase dentro del club. Queremos determinar precisamente esos datos y obtener una predicción. De esta manera quizá podamos recomendarle un horario o una actividad, o bien empezar a pensar si puede ser un cliente fiel o no.

Supongamos que disponemos de una tabla ampliada con información nueva de los clientes que incluye atributos nuevos, como las actividades deportivas principal y secundaria (*Act1* y *Act2*):

Cliente	Sexo	Horario	Renta	Edad	Años en el club	Act1	Act2
21	Mujer	Mañana	5.000.000	25	4	Stretch	TBC
212	Hombre	Mañana	10.000.000	65	1	Aeróbic	Steps
1	Mujer	Tarde	6.000.000	40	2	Aeróbic	Stretch
113	Mujer	Tarde	4.000.000	32	4	Yoga	Stretch
221	Mujer	Mañana	3.500.000	32	3	Yoga	Yoga
1234	Hombre	Tarde	1.200.000	18	4	Stretch	TBC
13	Hombre	Tarde	5.000.000	87	2	TBC	TBC

### Ejemplo de predicción

A la llegada de un cliente nuevo, podremos predecir según sus características, si tendrá tendencia a un tipo de ejercicios o de horario determinados, o el periodo durante el que será socio del club.

Cliente	Sexo	Horario	Renta	Edad	Años en el club	Act1	Act2
12	Hombre	Mañana	4.500.000	34	8	Stretch	TBC
111	Hombre	Tarde	2.300.000	56	9	Stretch	Stretch
1200	Hombre	Mañana	11.000.000	45	12	Steps	TBC
324	Mujer	Tarde	4.000.000	32	13	Steps	TBC
423	Hombre	Mañana	2.800.000	18	2	Yoga	TBC
567	Hombre	Tarde	27.000.000	19	3	Yoga	Stretch
777	Hombre	Tarde	12.000.000	45	12	TBC	Steps
3244	Hombre	Mañana	9.000.000	23	5	Steps	TBC
1223	Mujer	Tarde	23.450.000	67	6	Steps	Aeróbic
666	Hombre	Tarde	1.800.000	23	5	Aeróbic	Aeróbic
989	Mujer	Mañana	2.400.000	45	3	Aeróbic	Yoga
456	Mujer	Mañana	4.800.000	34	4	Aeróbic	Yoga
1325	Mujer	Mañana	5.700.000	34	8	Yoga	Steps

Supongamos también que definimos una función de similitud basada en los atributos que conocemos (*Sexo* y *Renta*, uno categórico y el otro numérico), que consista en combinar la distancia de Hamming y la euclidiana no normalizadas y normalizar dividiendo entre el número de atributos.

$$D_{sem} = (D_{ham} + D_{eu})/2$$

A partir de esta distancia podemos encontrar los vecinos del cliente nuevo. Cuando los hayamos encontrado, podemos consultar el valor que muestran para los atributos de interés (*Horario* y *Años en el club*). Aquí se nos plantean dos problemas: el número de vecinos que hay que considerar y el cálculo de los valores de los atributos desconocidos.

El primer problema admite varias respuestas. Evidentemente, cuantos más clientes queramos considerar, menos eficiente será el proceso de predicción. Por otra parte, tampoco son necesarios todos los vecinos; en general, con un número relativamente bajo de vecinos podemos hacer buenas predicciones. Todo depende de la calidad del conjunto de datos inicial y, en este caso, de la capacidad predictiva de dicho conjunto de datos.

El segundo problema se puede responder de dos maneras:

- Por mayoría. Se asignan a los atributos desconocidos los valores que con más frecuencia aparecen entre sus vecinos.
- Media. Se calcula el valor medio del atributo entre los vecinos (lo cual resultará problemático en el caso de las variables categóricas).

Intentamos realizar una predicción con los datos ficticios de que disponemos.

Calculando las distancias de la nueva cliente a los diferentes clientes, tenemos este orden (de más a menos próximo):

Ciente	Sexo	Horario	Renta	Edad	Años en el club	Act1	Act2
21	Mujer	Mañana	5.000.000,0	25	4	Stretch	TBC
456	Mujer	Mañana	4.800.000,0	34	4	Aeróbic	Yoga
1325	Mujer	Mañana	5.700.000,0	34	8	Yoga	Steps
1	Mujer	Tarde	6.000.000,0	40	2	Aeróbic	Stretch
113	Mujer	Tarde	4.000.000,0	32	4	Yoga	Stretch
324	Mujer	Tarde	4.000.000,0	32	13	Steps	TBC
221	Mujer	Mañana	3.500.000,0	32	3	Yoga	Yoga
989	Mujer	Mañana	2.400.000,0	45	3	Aeróbic	Yoga
1223	Mujer	Tarde	23.450.000,0	67	6	Steps	Aeróbic
1234	Hombre	Tarde	1.200.000,0	18	4	Stretch	TBC
666	Hombre	Tarde	1.800.000,0	23	5	Aeróbic	Aeróbic
111	Hombre	Tarde	2.300.000,0	56	9	Stretch	Stretch
423	Hombre	Mañana	2.800.000,0	18	2	Yoga	TBC
13	Hombre	Tarde	5.000.000,0	87	2	TBC	TBC
12	Hombre	Mañana	4.500.000,0	34	8	Stretch	TBC
3244	Hombre	Mañana	9.000.000,0	23	5	Steps	TBC
212	Hombre	Mañana	10.000.000,0	65	1	Aeróbic	Steps
1200	Hombre	Mañana	11.000.000,0	45	12	Steps	TBC
777	Hombre	Tarde	12.000.000,0	45	12	TBC	Steps
567	Hombre	Tarde	27.000.000,0	19	3	Yoga	Stretch

Parece que en este caso, uno de los atributos que más cuenta a la hora de definir la semejanza es el sexo. Intentemos hacer una predicción tomando los cinco vecinos más próximos:

Ciente	Sexo	Horario	Renta	Edad	Años en el club	Act1	Act2
21	Mujer	Mañana	5.000.000,0	25	4	Stretch	TBC
456	Mujer	Mañana	4.800.000,0	34	4	Aeróbic	Yoga
1325	Mujer	Mañana	5.700.000,0	34	8	Yoga	Steps
1	Mujer	Tarde	6.000.000,0	40	2	Aeróbic	Stretch
113	Mujer	Tarde	4.000.000,0	32	4	Yoga	Stretch
324	Mujer	Tarde	4.000.000,0	32	13	Steps	TBC

Observamos los hechos siguientes:


- 1) En el caso del horario, surge un empate; tenemos tantos vecinos que eligen horario de mañana como de tarde.
- 2) En el caso de años en el club, si elegimos la regla de la mayoría, parece que podemos predecir que la nueva clienta puede permanecer unos cuatro años en el club. Si aplicamos el criterio del valor medio, entonces podemos decir que permanecerá 5,8 años.
- 3) En cuanto a la actividad primaria que más puede interesarle, hay un empate entre yoga y aeróbic; en lo referente a la secundaria, si aplicamos otra vez la regla de la mayoría, tenemos otro empate entre TBC y *Stretch*.

Cuando se da esta situación, una posible solución sería experimentar con números de vecinos diferentes y asociar a cada número de vecinos un factor de confianza que indique el porcentaje de vecinos que cumplen el valor predicho.

### Actividad

5.1. Comparad las recomendaciones que se pueden hacer utilizando siete, ocho y nueve vecinos, y evaluad el grado de confianza de cada posibilidad.

Este hecho pone en evidencia que las predicciones que se puedan hacer serán tan buenas como los datos de que dispongamos. Tendrán la misma capacidad predictiva que la del conjunto de objetos que hayamos elegido para comparar.

Por lo tanto, es importante disponer de medios que permitan evaluar las características de los datos. 

### 5.2. Calidad de los modelos obtenidos

Hay varias maneras de evaluar la calidad del resultado de un proceso de agregación o de agrupación.


En principio, el **criterio de evaluación** depende de la aplicación a la que queramos destinar el resultado del proceso. Sin embargo, estableceremos algunas recomendaciones generales.

Un criterio de evaluación general es esperar obtener grupos muy cohesionados y, a la vez, muy diferentes entre sí. Esto quiere decir que los objetos de cada grupo son muy parecidos entre sí y al mismo tiempo muy diferentes de los objetos del resto de los grupos. Se trata de obtener agrupaciones con una alta similitud intragrupo y una baja similitud intergrupos.



Es preciso poder medir cada uno de estos aspectos:

- 1) Similitud intragrupo. Se trata de medir hasta qué punto son semejantes los objetos que el proceso de agregación ha incluido en un mismo grupo. Una medida típica es la varianza de los atributos considerados. Hay que analizar la matriz de covarianza de varios atributos.
- 2) Similitud intergrupos. Se puede considerar como la distancia media entre los centros de los diferentes grupos. Dados los resultados de dos procesos de agregación, hay que optar por el que presente más distancia entre centros.
- 3) Variantes. Otra manera de calcular la calidad de la agregación consiste en utilizar la medida de distancia que se ha empleado para construir los grupos y comparar la distancia media dentro de los *clusters* con respecto a la distancia media entre grupos.

El método de medida de la calidad de los modelos admite diferentes variaciones. 

#### Variaciones en la medida de la calidad de los modelos

El método de medida de la calidad del modelo puede variar, por ejemplo, en la manera de elegir las semillas iniciales o de calcular el centro siguiente.

A continuación, comentamos una nueva aplicación del principio de mínima longitud de descripción (MDL).


### 5.2.1. El principio de mínima longitud de descripción

Otras formas de medir la calidad de un modelo de agregación o de un modelo descriptivo en general, consisten en medir el grado de ajuste existente entre el modelo y los datos, o bien en estimar el grado de información que aporta el modelo con respecto a todos los modelos posibles que se pueden construir con los datos observados. Empezamos por este último, tomando la longitud de descripción como indicativo de la información que tiene el modelo.

El **principio de mínima longitud de descripción** (MDL, *Minimum Description Length*) se basa en la idea de que el mejor modelo que se puede obtener a partir de un conjunto de datos es el que, dado el modelo, minimiza la suma de la longitud de codificación del modelo y de los datos.

La segunda parte puede parecer extraña, pero debemos tener en cuenta que los datos se pueden conceptualizar como generados por el propio modelo. Es decir, este método estima que el mejor modelo es el que resulta más fácil de codificar a partir de los datos y penaliza los modelos que cumplen las dos condiciones siguientes:

- a) Son más complejos sobre los mismos datos.
- b) Necesitan más datos para ser codificados.

 Podéis consultar el método MDL en el subapartado 2.1.2 del módulo "Clasificación: árboles de decisión", de esta asignatura.

Vemos ahora cómo se aborda el problema de la codificación que se encuentra en la base del principio MDL. Supongamos que tenemos un conjunto de  $m$  observaciones (o casos)  $D$ . Cada observación está definida sobre el mismo dominio descrito por el mismo conjunto fijo de atributos  $X_1, \dots, X_n$ . Para simplificar, supondremos que cada atributo puede tomar un conjunto finito y determinado de valores. El conjunto de valores que puede tomar la variable  $X_i$  estará indexado desde 1 hasta  $q_i$ .

Un posible sistema de clasificación de los datos es el siguiente: damos una cadena binaria diferente a cada posible configuración de los datos (o del modelo, por ejemplo, de agregación). Todo el conjunto de datos  $D$  queda codificado por la concatenación de las  $m$  cadenas correspondientes a los  $m$  casos.

### Ejemplo de sistema de codificación de datos

Supongamos que tenemos tres variables  $X_1, \dots, X_3$ , cada una de las cuales puede adoptar dos valores: 'Presente' o 'Ausente'. Una codificación muy sencilla consiste en dar al valor 'Presente' el código 1 y al valor 'Ausente', el código 0. Si tenemos esta (pequeña) base de datos:

$X_1$	$X_2$	$X_3$	Codificación
Presente	Ausente	Presente	101
Presente	Presente	Presente	111
Ausente	Ausente	Presente	001

la codificación de todo el conjunto de datos corresponde a la secuencia 101111001.

Una manera de obtener la codificación global más corta posible (obtener la secuencia binaria más corta para representar el conjunto de datos) consiste en asignar los códigos más cortos a las observaciones más frecuentes. Para ello, se aplica el algoritmo de Huffman.

El algoritmo de Huffman consiste en ejecutar los pasos siguientes:

- 1) Tomar los dos valores del alfabeto de símbolos de codificación que tengan la frecuencia mínima. Asignar a esos dos símbolos, que tendrán la misma longitud y sólo diferirán en el último dígito, la secuencia de codificación más larga.
- 2) Combinar estos dos símbolos para crear un símbolo nuevo; calcular la probabilidad del símbolo nuevo y repetir el proceso.

Teniendo en cuenta que a cada paso se reduce el conjunto del alfabeto en una unidad, el algoritmo deberá asignar secuencias a todos los símbolos después de tantos pasos como símbolos haya en el alfabeto.

Supongamos que cada configuración  $c_i$  de la base de datos tiene una probabilidad  $p_i$ . El algoritmo de Huffman asigna a cada configuración  $c_i$  una codificación que posee una longitud aproximada de  $-\log_2(p_i)$ . Si tenemos  $m$


#### Asignación de secuencias

En el ejemplo de sistema de codificación de datos que hemos visto, cada combinación de valores  $x_1, x_2, x_3$  de las tres variables  $X_1, X_2$  y  $X_3$  es una configuración. La configuración del caso 1,  $c_1$ , es Presente-Ausente-Presente, o 101.

observaciones en la base de datos (tenemos  $m$  configuraciones), entonces la longitud de codificación de la base de datos es:

$$-m \sum_i p_i \log_2(p_i)$$

donde el sumatorio es sobre todas las configuraciones posibles.

Todas estas probabilidades no son conocidas, puesto que el modelo de agregación (de hecho, cualquier tipo de modelo elaborado a partir de los datos) sólo es una aproximación a la probabilidad de las configuraciones, que denotaremos por  $p_i$ . Cuanto más próxima esté la estimación  $\hat{p}_i$  a la probabilidad de los datos  $p_i$ , mejor es el modelo. 

Es importante observar que las probabilidades que nos ofrece el modelo nos permiten aproximar un cálculo de la longitud de codificación de los datos (la segunda parte del principio MDL).

Podemos utilizar las probabilidades  $\hat{p}_i$  que aparecen en el modelo como si fueran las probabilidades de aparición de las diferentes configuraciones en la población de configuraciones de donde proceden los datos. Es decir, supongamos que los datos se han generado en un proceso descrito por el modelo que hemos obtenido. Con esta aproximación y dado el modelo, obtenemos la longitud de descripción de los datos:

$$-m \sum_i p_i \log_2(\hat{p}_i)$$

Ahora podemos recurrir a un teorema con el fin de comparar ambas codificaciones.

### Teorema de Gibbs

El teorema de Gibbs establece que dadas dos secuencias finitas de números reales no negativos  $p_i$  y  $q_i$ , con  $i = 1, \dots, t$ , de manera que la suma de todos los números correspondientes de cada secuencia sea igual a 1, se cumple la desigualdad siguiente:

$$-m \sum_i p_i \log_2(p_i) \leq -m \sum_i p_i \log_2(\hat{p}_i)$$

La igualdad se da únicamente cuando para todo  $i$ ,  $p_i = \hat{p}_i$ . En el sumatorio consideramos que  $0 \log_2 0 = 0$ .

El **teorema de Gibbs** establece que la codificación que se obtiene utilizando las probabilidades estimadas (las que utiliza el modelo construido) es forzosamente de mayor longitud que la que se obtendría a partir las probabilidades verdaderas. También indica que con las probabilidades verdaderas se obtiene la longitud de codificación mínima.

El principio MDL indica que siempre debemos elegir el modelo que minimice la codificación de los datos. El problema es que desconocemos las probabilidades reales,  $p_i$ , y tenemos que estimarlas a partir de la frecuencia observada.

El principio MDL es lo suficientemente general para que podamos aplicarlo a una serie de modelos diferentes, tanto descriptivos (como son los modelos de agregación), como clasificadores y de otro tipo. Este principio se vuelve a ver al describir métodos de construcción de otros tipos de modelos. Para cada uno de estos métodos, lo que varía es la manera de codificar el modelo.

Podéis ver el método MDL en el subapartado 2.1.2 del módulo "Clasificación: árboles de decisión", de esta asignatura.

Veamos cómo podemos codificar un modelo de agregación. Recordemos que en la construcción de un modelo de agregación se obtienen una serie de grupos\* que en la mayoría de los casos representan una partición del conjunto inicial  $D$  de observaciones.

\* En inglés, *clusters*.

Supongamos que en el modelo tenemos  $k$  clusters. Cada *cluster* tiene una distribución de probabilidad para los valores de las observaciones que reúne. El mejor modelo será el que nos permita codificar los datos de manera más económica. Una posible forma consiste en codificar los centros de cada *cluster*. En principio, podemos considerar el centro como una observación ficticia sobre los atributos del dominio que recoge los valores medios de cada atributo.

### Ejemplo de codificación económica de los clientes de Hyper-Gym

Considerad la tabla siguiente:

Edad	Renta	Años en el club
34	5	4
45	9	14
56	7	10
19	2	2
34	4	6
67	4	7

Si esta tabla fuera la descripción de un *cluster* con una reducción de los datos de los clientes de Hyper-Gym, el centro del *cluster* sería la observación ficticia (para simplificar hemos puesto las medias sin decimal) obtenida calculando el valor medio de todos los atributos del *cluster*:

42	5	7
----	---	---

Para describir un modelo de agregación con  $k$  clusters, tendríamos  $k$  representaciones de este tipo.

una posible manera de proceder para encontrar la codificación (aunque no la única) consiste en efectuar los pasos siguientes para cada instancia:

- Indicar a qué *cluster* pertenece la instancia (lo cual requiere  $\log_2 k$  bits).
- Codificar los valores de los atributos con relación al centro del *cluster* (por ejemplo, expresando su diferencia o distancia absoluta).

### Ejemplo de codificación de los clientes de Hyper-Gym

Supongamos que el ejemplo de Hyper-Gym tuviera siete *clusters*. Entonces necesitaríamos tres bits para codificar a qué *cluster* pertenece cada observación (desde el *cluster* 000 al 111, 0 a 7 en binario).

Podéis ver el ejemplo de codificación económica de los clientes de Hyper-Gym en este subapartado.

Para cada observación del conjunto de datos original  $D$  tenemos una configuración de valores (*Edad*, *Renta*, *Años en el club*) dentro de todas las configuraciones posibles. El tercer elemento del *cluster* es:

56	7	10
----	---	----

Lo comparamos con el elemento centro:

42	5	7
----	---	---

y vemos que el valor absoluto de su diferencia con respecto al centro del *cluster* es:

14	4	4
----	---	---

Supongamos, para simplificar, que los valores de las tres variables oscilan entre 0 y 100. En este caso, necesitamos siete bits para cada variable, es decir, veintitún bits para cada configuración:

14	4	4
----	---	---

Si suponemos que el *cluster* al que pertenece es el 3, entonces la codificación sería:

Cluster	$X_1$	$X_2$	$X_3$
3	14	4	4
011	0001110	0000100	0000100

En el caso de los atributos categóricos, el procedimiento es más parecido al descrito. En efecto, cada valor recibe una codificación según con la distribución de probabilidad del atributo dentro del *cluster*.

De esta manera, debemos considerar como mejor la colección de *clusters* (modelo de agregación) que presente la mínima descripción de longitud.

## 5.2.2. Medidas de ajuste

Otra forma de medir la calidad de un modelo es ver la distancia que hay entre la distribución de probabilidad que indica el modelo que siguen sus atributos

y la distribución de probabilidad de la población a la que pertenecen los datos de la muestra.

### Ejemplo de medida de ajuste

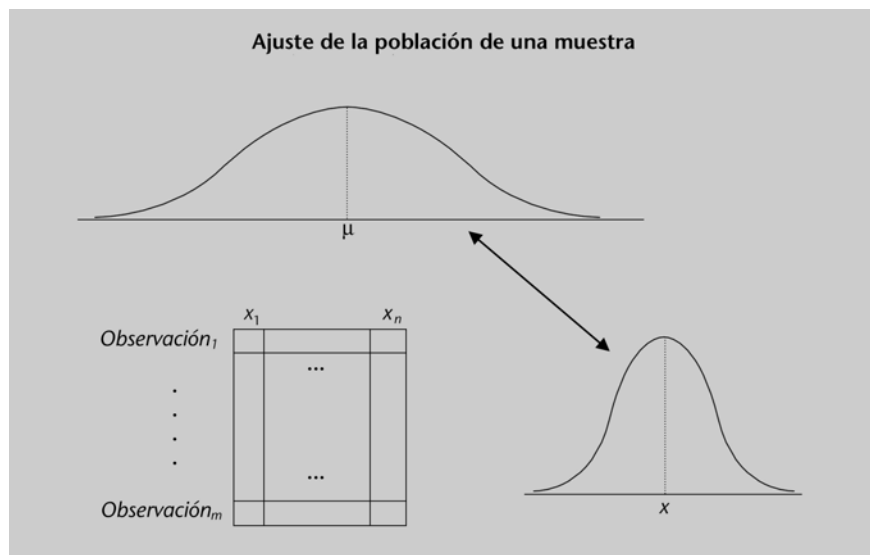
Supongamos un dominio definido sobre los atributos  $X_1, \dots, X_n$ . Si tenemos un conjunto de datos  $D$  formado por observaciones en el que cada observación es la concatenación de los valores de los atributos correspondientes  $x_1, \dots, x_n$ , podemos suponer que cada configuración  $x_1, \dots, x_n$  procede de una población cuyos valores siguen la distribución de probabilidad conjunta siguiente:

$$P(X_1, \dots, X_n)$$

Por otra parte, si hemos elaborado un modelo (de agregación, por ejemplo), podemos ver que los valores de los atributos  $X_1, \dots, X_n$  siguen una distribución como ésta:

$$\hat{P}(X_1, \dots, X_n)$$

Un criterio para elegir entre varios modelos es quedarse con el que indique que la distancia entre ambas distribuciones es mínima.



Ahora bien, ¿cómo medimos la distancia entre dos distribuciones, la que está implícita en el modelo y la que está implícita en los datos?

### Medida de Kullback-Leibler

Dadas dos distribuciones de probabilidad  $P$  y  $\hat{P}$ , la divergencia entre una y otra viene establecida por la expresión siguiente:

$$D_{kl}(P, \hat{P}) = \sum_i P(x_i) \log_2 \frac{P(x_i)}{\hat{P}(x_i)}$$

donde el sumatorio se hace sobre todas las configuraciones posibles.

La **divergencia de Kulback-Leibler** es siempre positiva, y sólo es cero cuando ambas distribuciones son idénticas.

Con el criterio de divergencia de Kulback-Leibler hay que utilizar el mismo tipo de conjunto de evaluación para poder comparar varios métodos de agregación elaborados a partir del mismo conjunto de datos y ver cuál es la divergencia que muestra cada modelo. Entonces nos quedamos con el que exhiba la divergencia mínima. !

Hay que matizar que si suponemos que cada *cluster* sigue su propia distribución (situación en la que todo el *cluster* debe conceptualizarse como una mixtura de distribuciones), la comparación no es tan directa.

Sin embargo, es interesante hacer hincapié en la relación entre el principio MDL y esta medida de divergencia. Si aplicamos el teorema de Gibbs, tenemos la propiedad KL-MDL.

! Podéis ver el principio MDL y el teorema de Gibbs en el subapartado 5.2.1 de este módulo.

La **propiedad KL-MDL** asegura que la longitud de codificación de los datos es una función monótona creciente de la divergencia entre la distribución definida por los datos y la definida por el modelo.

Por lo tanto, para poder evaluar y elegir entre modelos se puede utilizar tanto el principio MDL como la divergencia. !

## 6. Ponderación de los métodos de agregación

Para resumir brevemente los métodos de agregación, podemos mencionar sus ventajas e inconvenientes.

En primer lugar, las **ventajas de los métodos de agregación** son las siguientes:

- a) Resultados comprensibles, aunque no siempre directa o fácilmente interpretables. La presentación gráfica puede ser muy atractiva, pero es necesario que sea complementada con un análisis más profundo de las características de cada grupo y de las relaciones de cada grupo con respecto al resto.
- b) Resultado aplicable a varios tipos de datos. Aunque no todos, algunos de los métodos de agregación permiten combinar datos de varios tipos (numéricos, categóricos, etc.).
- c) Dimensionalidad. Permiten trabajar con dominios de alta dimensionalidad.
- d) Base para la predicción. Proporcionan un primer método para establecer predicciones. Como mínimo son un primer paso para emprender la construcción de modelos predictivos o clasificadores más especializados.

Los **inconvenientes de los métodos de agrupación** son los siguientes:

- a) Almacenamiento. Es un inconveniente que sufren especialmente los métodos que guardan información sobre los diferentes vecinos de cada caso.
- b) Escasa eficacia en la predicción. Por norma general, los métodos de agregación, que no están pensados para esta tarea predictiva, no dan buenos resultados en predicción.
- c) Sensibilidad en cuanto al orden de tratamiento de las observaciones. Esta dificultad es especialmente conflictiva en el caso de los métodos incrementales o aglomeradores.



## Resumen

Los métodos de agregación buscan en general un criterio para agrupar objetos parecidos. A efectos prácticos, sirven para tener una primera aproximación a la estructura del dominio y, aunque éste no es su objetivo principal, pueden desarrollar ciertas tareas de predicción. En efecto, si sabemos a qué grupo o *cluster* pertenece un objeto nuevo, podemos predecir sus características desconocidas a partir de las que muestran el resto de los objetos del grupo.

Los criterios para determinar la pertenencia de un objeto a un grupo se establecen a partir de medidas que pretenden reflejar la noción de *proximidad* en el espacio de observaciones.

Estas medidas se pueden basar en la idea de *distancia en el espacio de observaciones* o bien en la de *proximidad de las distribuciones de probabilidad conjunta* de los diferentes atributos que forman una observación.

Esto da lugar a dos grandes familias de métodos:

- Métodos basados en medidas de similitud o distancia. Por ejemplo, *k-means* y *k-nearest neighbours*.
- Métodos basados en propiedades de las distribuciones de probabilidad respectivas. Por ejemplo, COBWEB y Autoclass.

Una segunda manera de clasificar los métodos de agregación se relaciona con el tipo de procedimiento que se sigue para construir los modelos:


- Planos, en que se consideran todas las observaciones para construir el modelo y hay un único nivel de detalle en la construcción final, como en el *k-means*.
- Jerárquicos, donde los *clusters* se organizan en varios niveles de detalle o de generalización, como en el COBWEB.

En general, todos esos métodos siguen un esquema en el que, cuando aparece una observación nueva durante el proceso de construcción del modelo, es preciso decidir a qué grupo pertenece e incluirla en el mismo, o bien integrarla en el modelo como un grupo nuevo de un único elemento. Hay varios criterios para tomar la decisión, y las operaciones más habituales son la inclusión de una observación en un grupo, la fusión de grupos (de la que el anterior es un caso particular) y la división de grupos.

Para interpretar los modelos de agregación hay que estudiar las características de cada grupo y las de ese grupo con relación a los demás. Por lo general, se utiliza algún tipo de análisis estadístico sobre los valores medios y la dispersión de cada grupo y se intenta conocer qué combinaciones de atributos y valores se discriminan en cada grupo y cuáles son los atributos más relevantes.

La evaluación puede hacerse en función de la compacidad de cada grupo y su separación con respecto a los demás. En principio, son preferibles agregaciones en las que los componentes de cada grupo sean muy parecidos entre sí y muy diferentes del resto de los grupos. Asimismo, son preferibles las agregaciones en que haya más distancia entre grupos.

Las diferentes medidas de calidad utilizadas para construir agregaciones pretenden asegurar precisamente estas propiedades.

Otra manera de evaluar y comparar agregaciones es mediante otras propiedades que tienen en cuenta las propiedades de las distribuciones de probabilidad de la agregación obtenida. Por ejemplo, la medida MDL u otras medidas de ajuste. 

## Actividades

1. Acceded a la dirección de Internet que se da al margen y comparad las especificaciones de los diferentes *software* orientados a *clustering*.

Para hacer la actividad 1, acceded a la dirección <http://www.kdnuggets.com>.

2. Para el problema que se había propuesto en la actividad 1 del módulo “Extracción de conocimiento a partir de datos” de esta asignatura, ¿os sirven los métodos de agregación? ¿Qué método creéis que os resultaría más conveniente?

## Ejercicios de autoevaluación

1. Con los datos de la base de datos que os ofrecemos a continuación, resolved las actividades siguientes:

a) Calculad la distancia entre la observación 6 y el resto de los valores, teniendo en cuenta sólo los atributos numéricos:

Cliente	Sexo	Renta	Edad	Años en el club	Entrenador personal
1	Mujer	6.000.000	40	2	No
4	Hombre	3.200.000	35	6	No
6	Mujer	0	30	3	No
12	Hombre	4.500.000	34	8	No
18	Mujer	0	32	4	No
19	Hombre	3.000.000	37	3	No
20	Hombre	2.800.000	32	3	No
21	Mujer	0	32	4	No
22	Hombre	0	18	1	No
31	Mujer	4.000.000	30	1	No
33	Hombre	8.000.000	28	2	Sí
34	Mujer	1.200.000	55	8	No

Realizad el cálculo para las medidas de distancia siguientes:

- Valor absoluto de la diferencia
  - Cuadrado de la diferencia
  - Valor absoluto normalizado
  - Distancia euclidiana
  - Distancia de Hamming
- b) ¿Varía el resultado anterior si utilizáis la normalización por el máximo?
- c) ¿Varía el resultado anterior si discretizáis los valores?

2. Suponed que habéis encontrado los *clusters* siguientes, descritos por sus centros:

a)

Renta	Edad	Años en el club
6.000.000	40,3	2,8

b)

Renta	Edad	Años en el club
3.500.000	25,6	3,5

c)

Renta	Edad	Años en el club
1.000.000,56	45,6	8,5

d)

Renta	Edad	Años en el club
0	65	18,4

¿Cuáles son los *clusters* más alejados? ¿Y los más próximos? Intentad dar respuesta a estas preguntas utilizando las mismas distancias que en el ejercicio anterior.

## Bibliografía

**Béjar, J.** (1995). *Adquisición de conocimiento en dominios poco estructurados*. Tesis doctoral. Barcelona: Universidad Politécnica de Cataluña, Departamento de Lenguajes y Sistemas Informáticos.

**Cheeseman, P.; Stutz, J.** (1995). "Bayesian Classification (Autoclass): Theory and Results". En: U.M. Fayyad; G. Piatetsky-Sahpiro; P. Smyth; R. Uthurusamy (eds.). *Advances in Knowledge Discovery and Data Mining* (págs. 153-180). AAAI Press.

**Duran, B.S.; Odel, P.L.** (1974). "Cluster Analysis: a Survey". *Lecture Notes in Economics and Mathematical Systems* (vol. 100). Springer-Verlag.

**Fisher, D.** (1987). "Knowledge Acquisition via Incremental Conceptual Clustering". *Machine Learning* (vol. 2, núm. 2, págs. 139-172).

**Fisher, D.; Xu, Li; Zard, N.** (1992). "Ordering Effects in Clusterings". *Proceedings of the Ninth International Workshop on Machine Learning* (págs. 163-168).

**Hart, P.** (1967). "The Condensed Nearest Neighbour Rule". *Transactions of Information Theory* (núm. 14, págs. 515-516). IT.

**Hartigan, J.; Wong, M.** (1979). "A k-means Clustering Algorithm". *Applied Statistics* (núm. 28, vol. 1).

**Michalski, R.S.; Stepp, R.E.** "Clustering". *Encyclopedia of Artificial Intelligence* (págs. 168-176).

**Michalski, R.S.; Stepp, R.E.** "Learning from Observation: Conceptual Clustering". En: R.S. Michalski *Machine Learning: an Artificial Intelligence Approach* (págs. 331-363).

**Roure, J.** (1998). "Robust Incremental Clustering with Bad Instance Orderings: a New Strategy". En: H. Coelho (ed.). *Progress in Artificial Intelligence: IBERAMIA '98. Sixth Iberoamerican Conference on Artificial Intelligence* (págs. 136-147). Lisboa: Springer Verlag.

