Análisis de Cluster y Arboles de Clasificación

Introducción

El análisis de cluster es una técnica cuya idea básica es agrupar un conjunto de observaciones en un número dado de *clusters* o grupos. Este agrupamiento se basa en la idea de *distancia* o similitud entre las observaciones, que se se estudió en el tema de Multidimensional Scaling.

La obtención de dichos clusters depende del criterio o distancia considerados; así, por ejemplo, una baraja de cartas españolas se podría dividir de distintos modos: en cuatro clusters (los cuatro palos), en ocho clusters (los cuatro palos y según sean figuras o números), en dos clusters (figuras y números). Es decir, todo depende de lo que consideremos como *similar*.

El número posible de combinaciones de grupos y de elementos que integran los posibles grupos se hace intratable desde el punto de vista computacional, aún con un número escaso de observaciones.

Se hace necesario, pues, encontrar métodos o algoritmos que infieran el número y componentes de los clusters más aceptable, aunque no sea el óptimo absoluto.

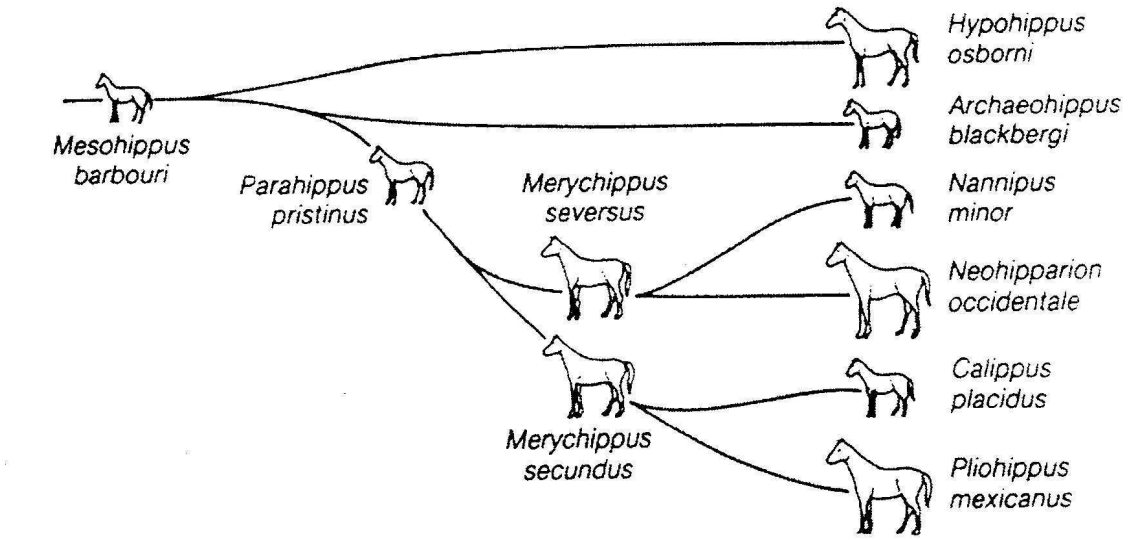
Métodos de cluster jerárquicos

En la práctica, no se pueden examinar todas las posibilidades de agrupar los elementos, incluso con los ordenadores más rápidos. Una solución se encuentra en los llamados

métodos jerárquicos. Se tienen dos posibles formas de actuar:

Métodos jerárquicos aglomerativos: se comienza con los objetos o individuos de modo individual; de este modo, se tienen tantos clusters iniciales como objetos. Luego se van agrupando de modo que los primeros en hacerlo son los más similares y al final, todos los subgrupos se unen en un único cluster.

Métodos jerárquicos divididos: se actúa al contrario. Se parte de un grupo único con todas las observaciones y se van dividiendo según lo *lejanos* que estén.



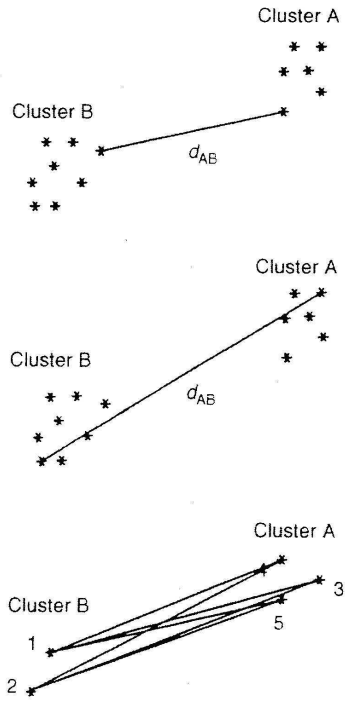
En cualquier caso, de ambos métodos se deriva un *dendograma*, que es un gráfico que ilustra cómo se van haciendo las subdivisiones o los agrupamientos, etapa a etapa.

Consideramos aquí los métodos aglomerativos con diferentes métodos de unión (*linkage methods*). Los más importantes son:

(*i*) Mínima distancia o vecino más próximo.

(*ii*) Máxima distancia o vecino más lejano.

(*iii*) Distancia media (average distance).



Definidas las distancias anteriores, se puede considerar el algoritmo básico, dados N objetos o individuos:

1. Empezar con N clusters (el número inicial de elementos) y una matriz $N \times N$ simétrica de distancias o similitudes. $D = [d_{ik}]_{ik}$.

2. Dentro de la matriz de distancias, buscar aquella entre los clusters U y V (más próximos, más distantes o en media más próximos) que sea la menor entre todas, d_{uv}.

3. Juntar los clusters U y V en uno solo. Actualizar la matriz de distancias:

   (i) Borrando las filas y columnas de los clusters U y V.

   $ii)$ Formando la fila y columna de las distancias del nuevo cluster (UV) al resto de clusters.

4. Repetir los pasos (2) y (3) un total de $(N-1)$ veces.

Al final, todos los objetos están en un único cluster cuando termina el algoritmo. Además, se guarda la identificación de los clusters que se van uniendo en cada etapa, así como las distancias a las que se unen. Finalmente se construye un *dendograma*.

Ejemplo con *mínima distancia*:

Sea la matriz de distancias entre 5 objetos la dada por:

$$D = [d_{ik}]_{ik} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} 0 & & & & \\ 9 & 0 & & & \\ 3 & 7 & 0 & & \\ 6 & 5 & 9 & 0 & \\ 11 & 10 & \mathbf{2} & 8 & 0 \end{bmatrix} \end{array}$$

Cada uno de los objetos comienza siendo un cluster. Como $\text{mín}_{i,k}\, d_{ik} = d_{53} = 2$ los objetos 3 y 5 se unen para formar el cluster (35). Para construir el siguiente nivel, calculo la distancia entre el cluster (35) y los restantes objetos 1, 2 y 4. Así:

$$\begin{aligned} d_{(35),1} &= \text{mín}\{d_{31}, d_{51}\} = \text{mín}\{3, 11\} = 3 \\ d_{(35),2} &= \text{mín}\{d_{32}, d_{52}\} = \text{mín}\{7, 10\} = 7 \\ d_{(35),4} &= \text{mín}\{d_{34}, d_{54}\} = \text{mín}\{9, 8\} = 8 \end{aligned}$$

Reconstruyo la matriz de distancias:

$$D = [d_{ik}]_{ik} = \begin{array}{c} \\ (35) \\ 1 \\ 2 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} (35) & 1 & 2 & 4 \end{array} \\ \begin{bmatrix} 0 & & & \\ \mathbf{3} & 0 & & \\ 7 & 9 & 0 & \\ 8 & 6 & 5 & 0 \end{bmatrix} \end{array}$$

Cojo la menor distancia, $d_{(35),1} = 3$ y junto, así, el cluster (35) con el 1.

Calculo ahora las distancias del nuevo cluster a los dos elementos que quedan:

$$\begin{aligned} d_{(351),2} &= \text{mín}\{d_{(35),2}, d_{12}\} = \text{mín}\{7, 9\} = 7 \\ d_{(351),4} &= \min\{d_{(35),4}, d_{14}\} = \min\{8, 6\} = 6 \end{aligned}$$

La matriz de distancias queda como:

$$D = [d_{ik}]_{ik} = \begin{matrix} \\ (351) \\ 2 \\ 4 \end{matrix} \begin{matrix} \begin{matrix} (351) & 2 & 4 \end{matrix} \\ \begin{bmatrix} 0 & & \\ 7 & 0 & \\ 6 & \mathbf{5} & 0 \end{bmatrix} \end{matrix}$$

La mínima distancia se alcanza entre los clusters 2 y 4 $d_{24} = 5$. Se obtienen así dos clusters: (351) y (24). La distancia que los separa es:

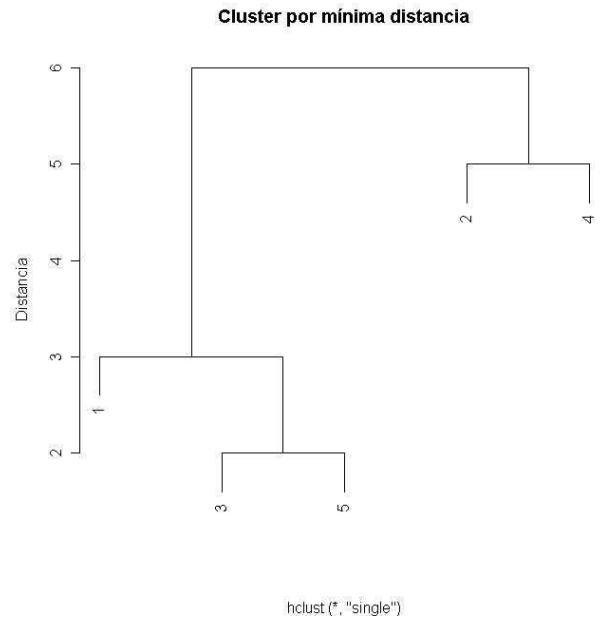
$$d_{(351),(24)} = \text{mín}\{d_{(351),2}, d_{(351),4}\} = \text{mín}\{7, 6\} = 6$$

Así, la matriz de distancias queda como:

$$D = [d_{ik}]_{ik} = \begin{matrix} \\ (351) \\ (24) \end{matrix} \begin{matrix} \begin{matrix} (351) & (24) \end{matrix} \\ \begin{bmatrix} 0 & \\ 6 & 0 \end{bmatrix} \end{matrix}$$

Cuando la distancia es iguala 6, todos los objetos se unen en un único cluster.

Se pueden dibujar dendogramas:



Este tipo de distancia no funciona bien cuando los objetos están próximos.

Se obtienen dendogramas similares si se utiliza la distancia máxima, o la distancia media, aunque las distancias a las que se van uniendo los objetos en los clusters varían en cada caso.

Problemas

- Las fuentes de error y variación no entran en consideración con los métodos jerárquicos. Esto implica una gran sensibilidad a observaciones anómalas o *outliers*.

- Si un objeto se ha colocado erróneamente en un grupo al principio del proceso, ya no se puede arreglar en una etapa posterior.

- Un sistema de trabajo conveniente es usar varias distancias o similitudes con los mismos objetos y observar si se mantienen los mismos clusters o grupos. Así, se comprueba la existencia de grupos naturales.

Estos métodos se pueden usar para clasificar no sólo observaciones, sino también variables usando como medida de similitud algún coeficiente de correlación.

Métodos no jerárquicos

Se usan para agrupar objetos, pero no variables, en un conjunto de k clusters ya predeterminado. No se tiene que especificar una matriz de distancias ni se tienen que almacenar las iteraciones. Todo esto permite trabajar con un número de datos mayor que en el caso de los métodos jerárquicos.

Se parte de un conjunto inicial de clusters elegidos al azar, que son los *representantes* de todos ellos; luego se van cambiando de modo iterativo. Se usa habitualmente el método de las k-medias.

Método de las k-medias

Es un método que permite asignar a cada observación el cluster que se encuentra más próximo en términos del centroide (media). En general, la distancia empleada es la euclídea.

Pasos:

1. Se toman al azar k clusters iniciales.

2. Para el conjunto de observaciones, se vuelve a calcular las distancias a los centroides de los clusters y se reasignan a los que estén más próximos. Se vuelven a recalcular los centroides de los k clusters después de las reasignaciones de los elementos.

3. Se repiten los dos pasos anteriores hasta que no se produzca ninguna reasignación, es decir, hasta que los elementos se estabilicen en algún grupo.

Usualmente, se especifican k centroides iniciales y se procede al paso (2) y, en la práctica, se observan la mayor parte de reasignaciones en las primeras iteraciones.

Ejemplo Supongamos dos variables x_1 y x_2 y 4 elementos: A, B, C, D. con la siguiente tabla de valores:

	x_1	x_2
A	5	3
B	-1	1
C	1	-2
D	-3	-2

Se quiere dividir estos elementos en dos grupos ($k = 2$).

De modo arbitrario, se dividen los elementos en dos clusters (AB) y (CD) y se calculan los centroides de los dos clusters.

Cluster (AB) :

$\bar{x}_1$	$\bar{x}_2$
$\frac{5-1}{2} = 2$	$\frac{3+1}{2} = 2$

Cluster (CD) :

$\bar{x}_1$	$\bar{x}_2$
$\frac{1-3}{2} = -1$	$\frac{-2-2}{2} = -2$

En el paso (2), calculamos las distancias euclídeas de cada observación al grupo de centroides y reasignamos cada una al grupo más próximo. Si alguna observación se mueve de grupo, hay que volver a calcular los centroides de los grupos. Así, las distancias son:

$$\begin{aligned} d^2(A,(AB)) &= (5-2)^2 + (3-2)^2 = 10 \\ d^2(A,(CD)) &= (5+1)^2 + (3+2)^2 = 61 \end{aligned}$$

Como A está más próximo al cluster (AB) que al cluster (CD), no se reasigna.

Se hace lo mismo para el elemento B:

$$\begin{aligned} d^2(B,(AB)) &= (-1-2)^2 + (1-2)^2 = 10 \\ d^2(B,(CD)) &= (-1+1)^2 + (1+2)^2 = 9 \end{aligned}$$

Por lo cual, el elemento B se reasigna al cluster (CD) dando lugar al cluster (BCD). A continuación, se vuelven a calcular los centroides:

Cluster	$\bar{x}_1$	$\bar{x}_2$
A	5	3
(BCD)	-1	-1

Nuevamente, se vuelven a calcular las distancias para cada observación para ver si se producen cambios con respecto a los nuevos centroides:

	A	(BCD)
A	0	52
B	40	4
C	41	5
D	89	5

Como no se producen cambios, entonces la solución para $k = 2$ clusters es: A y (BCD).

Si se quiere comprobar la estabilidad de los grupos, es conveniente volver a correr el algoritmo con otros clusters iniciales (una nueva partición inicial).

Una vez considerados los clusters finales, es conveniente interpretarlos; para ello, se pueden cruzar con otras variables categóricas o se pueden ordenar de modo que los objetos del primer cluster aparezcan al principio y los del último cluster al final.

Tablas de análisis de la varianza

El objetivo que se persigue al formar los clusters es que los centroides estén lo más separados entre sí como sea posible y que las observaciones dentro de cada cluster estén muy próximas al centroide. Lo anterior se puede medir con el estadístico de la F de Snedecor:

$$F_{n,m} = \frac{\chi^2_n / n}{\chi^2_m / m},$$

y equivale al cociente de dos distribuciones chi-cuadrado divididas entre sus grados de libertad.

El estadístico F se calcula, así, como un cociente de *medias de cuadrados*. En el caso del análisis de cluster:

$$F = \frac{\text{medias de cuadrados entre clusters}}{\text{medias de cuadrados dentro de clusters}}$$

Si $F > 1$, las distancias entre los centroides de los grupos son mayores que las distancias de los elementos dentro de los grupos. Esto es lo que se pretende para que los clusters estén suficientemente diferenciados entre sí.

Problemas que surgen al fijar *k* clusters iniciales

(*i*) Si dos centroides iniciales caen por casualidad en un único cluster natural, entonces los clusters que resultan están poco diferenciados entre sí.

(*ii*) Si aparecen outliers, se obtiene por lo menos un cluster con sus objetos muy dispersos.

(iii) Si se imponen previamente k clusters puede dar lugar a grupos artificiales o bien a juntar grupos distintos.

Una posible solución es considerar varias elecciones del número k de clusters comparando luego sus coeficientes de la F de Snedecor.

Ejemplos

Se considera una muestra de los años de vida esperados por país, edad y sexo procedentes de Keyfitz y Flieger (1971).

	m0	m25	m50	m75	w0	w25	w50	w75
Algeria	63.00	51.00	30.00	13.00	67.00	54.00	34.00	15.00
Cameroon	34.00	29.00	13.00	5.00	38.00	32.00	17.00	6.00
Madagascar	38.00	30.00	17.00	7.00	38.00	34.00	20.00	7.00
Mauritius	59.00	42.00	20.00	6.00	64.00	46.00	25.00	8.00
Reunion	56.00	38.00	18.00	7.00	62.00	46.00	25.00	10.00
Seychelles	62.00	44.00	24.00	7.00	69.00	50.00	28.00	14.00
South Africa(C)	50.00	39.00	20.00	7.00	55.00	43.00	23.00	8.00
South Africa(W)	65.00	44.00	22.00	7.00	72.00	50.00	27.00	9.00
Tunisia	56.00	46.00	24.00	11.00	63.00	54.00	33.00	19.00
Canada	69.00	47.00	24.00	8.00	75.00	53.00	29.00	10.00
Costa Rica	65.00	48.00	26.00	9.00	68.00	50.00	27.00	10.00
Dominican Rep	64.00	50.00	28.00	11.00	66.00	51.00	29.00	11.00
El Salvador	56.00	44.00	25.00	10.00	61.00	48.00	27.00	12.00
Greenland	60.00	44.00	22.00	6.00	65.00	45.00	25.00	9.00
Grenada	61.00	45.00	22.00	8.00	65.00	49.00	27.00	10.00
Guatemala	49.00	40.00	22.00	9.00	51.00	41.00	23.00	8.00
Honduras	59.00	42.00	22.00	6.00	61.00	43.00	22.00	7.00
Jamaica	63.00	44.00	23.00	8.00	67.00	48.00	26.00	9.00
Mexico	59.00	44.00	24.00	8.00	63.00	46.00	25.00	8.00
Nicaragua	65.00	48.00	28.00	14.00	68.00	51.00	29.00	13.00
Panama	65.00	48.00	26.00	9.00	67.00	49.00	27.00	10.00
Trinidad(62)	64.00	63.00	21.00	7.00	68.00	47.00	25.00	9.00
Trinidad (67)	64.00	43.00	21.00	6.00	68.00	47.00	24.00	8.00
United States (66)	67.00	45.00	23.00	8.00	74.00	51.00	28.00	10.00
United States (NW66)	61.00	40.00	21.00	10.00	67.00	46.00	25.00	11.00
United States (W66)	68.00	46.00	23.00	8.00	75.00	52.00	29.00	10.00
United States (67)	67.00	45.00	23.00	8.00	74.00	51.00	28.00	10.00
Argentina	65.00	46.00	24.00	9.00	71.00	51.00	28.00	10.00
Chile	59.00	43.00	23.00	10.00	66.00	49.00	27.00	12.00
Columbia	58.00	44.00	24.00	9.00	62.00	47.00	25.00	10.00
Ecuador	57.00	46.00	28.00	9.00	60.00	49.00	28.00	11.00

Se considera otra muestra de 48 objetos de cerámica romana donde se miden diferentes tipos de oxidación:

	AL2O3	FE2O3	MGO	CAO	NA2O	K2O	TIO2	MNO	BAO
1	1.76	1.11	0.30	0.46	0.50	1.02	1.29	0.48	1.07
2	1.58	0.85	0.25	0.49	0.50	0.97	1.27	0.41	1.29
3	1.70	0.89	0.27	0.45	0.50	0.98	1.26	0.54	1.00
4	1.58	0.85	0.23	0.44	0.50	0.97	1.28	0.39	1.36
5	1.66	0.84	0.27	0.53	0.54	0.99	1.19	0.38	1.36
6	1.76	0.87	0.31	0.51	0.31	1.04	1.26	0.44	1.21
7	1.54	0.82	0.27	1.01	0.41	1.02	1.22	0.41	1.36
8	1.68	0.86	0.31	0.58	0.35	1.07	1.23	0.44	1.21
9	1.48	0.83	0.24	0.41	0.48	1.04	1.19	0.38	1.21
10	1.36	0.80	0.25	0.44	0.41	0.97	1.17	0.34	0.86
11	1.28	0.68	0.22	0.38	0.16	0.72	0.96	0.21	0.86
12	1.36	0.79	0.24	0.86	0.25	0.96	1.12	0.34	1.14
13	1.38	0.82	0.24	0.84	0.30	0.96	1.10	0.49	1.14
14	1.60	0.91	0.30	0.48	0.58	1.00	1.19	0.56	1.43
15	1.57	0.91	0.28	0.49	0.58	0.93	1.21	0.58	1.43
16	1.48	0.89	0.29	0.47	1.04	1.06	1.23	0.69	1.36
17	1.74	0.91	0.35	0.51	0.48	1.01	1.26	0.50	1.29
18	1.58	0.92	0.27	0.76	0.66	0.98	1.22	0.57	1.64
19	1.77	0.88	0.31	0.48	0.16	1.05	1.26	0.44	1.07
20	1.68	0.87	0.29	0.40	0.15	1.00	1.19	0.22	1.21
21	1.66	0.85	0.29	0.47	0.23	1.00	1.15	0.41	1.21
22	1.35	0.81	0.64	0.09	0.64	1.35	1.01	0.99	1.36
23	1.29	0.82	0.51	0.07	0.21	1.32	0.99	0.89	1.43
24	1.36	0.82	0.58	0.08	0.25	1.39	1.04	0.77	1.36
25	1.07	0.74	0.84	0.09	0.18	1.24	0.88	0.54	0.64
26	1.29	0.82	0.80	0.12	0.25	1.37	0.91	0.62	1.50
27	1.02	0.73	0.52	0.10	0.28	1.08	0.85	0.67	0.71
28	0.94	0.50	0.64	0.12	0.23	1.06	0.76	0.92	1.21
29	1.08	0.67	0.88	0.10	0.20	1.18	0.83	0.51	1.07
30	1.04	0.64	0.67	0.17	0.38	1.28	0.81	0.49	1.14
31	1.25	0.80	1.08	0.16	0.25	1.45	0.88	1.01	1.21
32	1.16	0.71	0.85	0.13	0.68	1.48	0.90	0.98	1.07
33	1.22	0.77	0.82	0.18	0.30	1.56	0.92	0.58	1.21
34	1.08	0.63	0.56	0.17	0.07	1.44	0.72	0.68	1.07
35	1.10	0.63	0.59	0.17	0.05	1.48	0.76	0.52	0.93
36	1.71	0.15	0.10	0.02	0.04	0.62	0.83	0.01	1.00
37	1.48	0.28	0.09	0.01	0.05	0.62	1.65	0.01	1.00
38	1.68	0.17	0.10	0.01	0.07	0.67	1.18	0.01	1.14
39	1.68	0.22	0.10	0.01	0.05	0.64	1.42	0.04	1.57
40	1.94	0.18	0.11	0.04	0.13	0.75	1.62	0.01	1.14
41	1.65	0.13	0.08	0.03	0.07	0.66	1.01	0.01	0.93
42	1.71	0.13	0.10	0.03	0.06	0.67	1.14	0.04	1.36
43	1.56	0.11	0.08	0.01	0.06	0.56	1.17	0.02	0.93
44	1.38	0.32	0.10	0.02	0.06	0.68	1.72	0.02	1.07
45	1.79	0.19	0.09	0.06	0.04	0.56	1.33	0.04	1.29

Ejemplo de esperanza de vida

```
# Se dibujan los dendogramas segun tres tipos de linkages
par(mfrow=c(1,3))

plclust(hclust(dist(life),method="single"),labels=row.names(life),
ylab="Distancia")
title("(a) Minima distancia")

plclust(hclust(dist(life),method="complete"),labels=row.names(life),
ylab="Distancia")
title("(b) Maxima distancia")

plclust(hclust(dist(life),method="average"),labels=row.names(life),
ylab="Distancia")
title("(c) Distancia media")

# Se determinan los paises que pertenecen a cada cluster
# usando el linkage del maximo, cortando a una distancia de 21
cuantos <- cutree(hclust(dist(life),method="complete"),h=21)
pais.clus <- lapply(1:5, function(eso){row.names(life)[cuantos==eso]})
pais.clus

# Se calculan las medias de cada variable dentro de cada cluster
pais.medias <- lapply(1:5,function(eso){apply(life[cuantos==eso,]
,2,mean)})
pais.medias
```

```
# Se dibujan los cruces de variables con el cluster de
# pertenencia identificado
pairs(life,panel= function(x,y){text(x,y,cuantos)})
```

Ejemplo de cerámicas romanas

```
# Para que las escalas de las variables sean iguales, se
# divide cada valor entre el rango de las variables: (max-min)
rge <- apply(cacharros,2,max)-apply(cacharros,2,min)
cacharros <- sweep(cacharros,2,rge,FUN="/")
n <- length(cacharros[,1])

# Se calculan las sumas de cuadrados dentro de grupos para todos los datos
# Se calcula la suma de cuadrados dentro de grupos con 1 solo grupo
scd1 <- (n-1)*sum(apply(cacharros,2,var))

# Se calcula la suma de cuadrados dentro de grupos con 2 a 6  grupos
scd <- numeric(0)
for(i in 2:6) {W <- sum(kmeans(cacharros,i)$withinss)
         scd <- c(scd,W) }

# Se juntan los resultados de 1 grupo con los de 2:6 grupos
scd <- c(scd1,scd)

# Se dibujan las sumas de cuadrados dentro de grupos frente
# al numero de grupos
plot(1:6,scd,type="l",xlab="Numero de grupos",ylab="Suma de cuadrados
dentro de grupos", lwd=2)
```

```
# El resultado mejor es con 2 o 3 grupos
cacharros.kmedia <- kmeans(cacharros,3)
cacharros.kmedia

# Los resultados anteriores son para los datos estandarizados.
# Para calcular los resultados sobre los resultados reales:
lapply(1:3, function(eso){apply(cacharros[cacharros.kmedia$cluster==eso,]
,2,mean)} )
```

Arboles de Clasificación

Los métodos basados en árboles (o *árboles de decisión*) son bastante populares en data mining, pudiéndose usar para clasificación y regresión. Estos métodos se derivan de una metodología previa denominada *automatic interaction detection*.

Son útiles para la exploración inicial de datos y apropiados cuando hay un número elevado de datos, y existe incertidumbre sobre la manera en que las variables explicativas deberían introducirse en el modelo. Sin embargo, no constituyen una herramienta demasiado precisa de análisis.

En conjuntos pequeños de datos es poco probable que revelen la estructura de ellos, de modo que su mejor aplicación se encuentra en grandes masas de datos donde pueden revelar formas complejas en la estructura que no se pueden detectar con los métodos convencionales de regresión.

Problemas donde los árboles de clasificación se pueden usar

1. Regresión con una variable dependiente continua.
2. Regresión binaria.
3. Problemas de clasificación con categorías múltiples ordinales.
4. Problemas de clasificación con categorías múltiples nominales.

Ventajas de los árboles de clasificación

1. Los resultados son invariantes por una transformación de monótona de las variables explicativas.
2. La metodología se adapta fácilmente en situaciones donde aparecen datos missing, sin necesidad de eliminar la observación completa.
3. Están adaptados para recoger el comportamiento *no aditivo*, de manera que las interacciones se incluyen de manera automática.

4. Incluye modelos de regresión así como modelos de clasificación generales que se pueden aplicar de manera inmediata para diagnosis.

Desventajas

1. El árbol final puede que no sea óptimo. La metodología que se aplica sólo asegura cada subdivisión es óptima.

2. Las variables predictoras (independientes) continuas se tratan de manera ineficiente como variables categóricas discretas.

3. Las interacciones de orden menor no preceden a las interacciones de orden mayor.

4. Los árboles grandes tienen poco sentido intuitivo y las predicciones tienen, a veces, cierto aire de *cajas negras*.

Suponemos que se dispone de una muestra de entrenamiento que incluye la información del grupo al que pertenece cada caso y que sirve para construir el criterio de clasificación.

Se comienza con un nudo inicial y nos preguntamos cómo dividir el conjunto de datos disponibles en dos partes más homogéneas utilizando una de las variables. Esta variable se escoge de modo que la partición de datos se haga en dos conjuntos lo más homogéneos posibles.

Se elige, por ejemplo, la variable x_1 y se determina un punto de corte, por ejemplo c de modo que se puedan separar los datos en dos conjuntos: aquellos con $x_1 \leq c$ y los que tienen $x_1 > c$.

De este nodo inicial saldrán ahora dos: uno al que llegan las observaciones con $x_1 \leq c$ y otro al que llegan las observaciones con $x_1 > c$.

En cada uno de estos nodos se vuelve a repetir el proceso de seleccionar una variable y un punto de corte para dividir la muestra en dos partes más homogéneas. El proceso termina cuando se hayan clasificado todas las observaciones correctamente en su grupo.

Para construir un árbol se han de tomar las siguientes decisiones:

(*i*) Seleccionar las variables y sus puntos de corte para hacer las divisiones.

(*ii*) Cuándo se considera que un nudo es terminal y cuándo se continúa dividiendo.

(*iii*) La asignación de las clases a los nudos terminales.

Se puede asociar a cada nudo el subconjunto de observaciones que pasa por él. Para decidir qué variable va a utilizarse para hacer la partición en un nudo se calcula primero la proporción de observaciones que pasan por el nudo para cada uno de los grupos. Si se denomina a los nudos como $t = 1, 2, \ldots, T$ y $p(g|t)$ a las probabilidades de que las observaciones que lleguen al nudo t pertenezcan a cada una de las clases, se define la *impureza* del nudo t como

$$I(t) = -\sum_{g=1}^{G} p(g|t) \cdot \log p(g|t)$$

que es una medida de la entropía o diversidad. Esta es máxima cuando $p(g|t) = 1/G$.

La variable que se introduce en un nudo es la que minimiza la heterogeneidad o impureza que resulta de la división en el nudo.

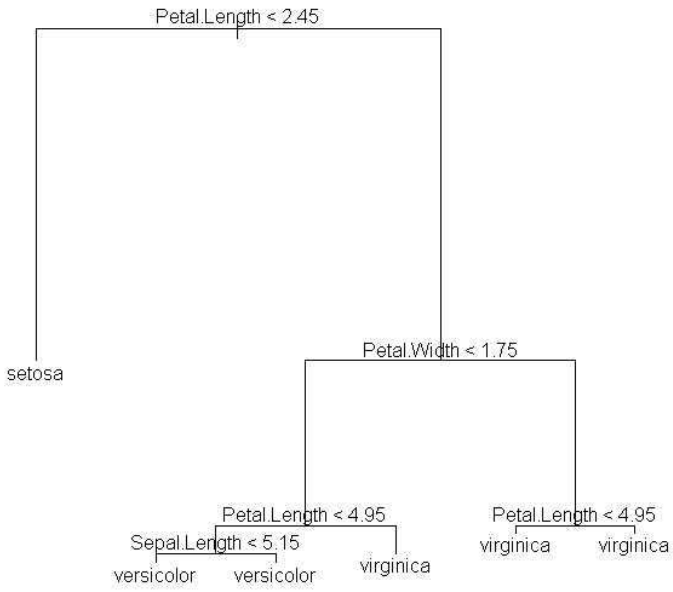
La clasificación de las observaciones en los nudos terminales se hace asignando todas las observaciones del nudo al grupo más probable en ese nudo, es decir, el grupo con máxima $p(g|t)$. Si la impureza del nudo es cero, todas las observaciones pertenecerían al mismo nudo, en caso contrario puede haber cierto error de clasificación. Cuando el número de variables es grande, el árbol puede contener un número excesivo de nudos por lo que se hace necesario definir procedimientos de *poda* o simplificación del mismo.

Ejemplo: Flores el género Iris.

```
library(tree)
library(datasets)


iris
```

```
iris.tr <- tree(Species ~., iris)
iris.tr
summary(iris.tr)
plot(iris.tr)
text(iris.tr)
```



Ejemplo: 4601 emails, de los que se identifican como spam a 1813 (mira la librería DAAG de `R`).

Las variables explicativas son

- `crl.tot`, longitud total de las palabras que están en mayúsculas;
- `dollar`, frecuencia del símbolo `$`, en términos del porcentaje respecto de caracteres;

- `bang`, frecuencia del símbolo `!`, en términos del porcentaje respecto de caracteres;
- `money` frecuencia de la palabra *money*, en términos del porcentaje respecto de caracteres;
- `n000`, frecuencia de la cadena *000*, en términos del porcentaje respecto de caracteres;
- `make`, frecuencia de la palabra *make*, en términos del porcentaje respecto de caracteres.

La variable dependiente es `yesno`, que es `n` para no-spam y `y` para spam.

Se aplica un árbol de clasificación usando las 6 variables como predictoras.

```
library(DAAG)
library(rpart)
data(spam7)
attach(spam7)
spam.tree <- rpart(formula = yesno ~ crl.tot + dollar + bang +
money + n000 + make, method="class", data=spam7)
jpeg(file="image6.jpg",pointsize=10,width=600,height=600,quality=75)
plot(spam.tree)
text(spam.tree)
options(digits=5)
dev.off()
printcp(spam.tree)
```

- Es conveniente encontrar un tamaño óptimo del árbol de modo que se encuentre el equilibrio entre la complejidad del árbol y el ajuste a las observaciones.
- En la librería `rpart`, se define el *parámetro de complejidad* `cp`. Se obtiene un árbol óptimo en función de este parámetro.

```
spam7a.tree <- rpart(formula = yesno ~ crl.tot + dollar +
bang + money + n000 + make, method="class", data=spam7,cp=0.001)
plotcp(spam7a.tree)
printcp(spam7a.tree)
```

Se considera la *poda* del árbol óptimo, ajustando el árbol que tenga menor valor de cp.

```
spam7b.tree <- prune(spam7a.tree,
cp=spam7a.tree$cptable[which.min(spam7a.tree$cptable[,"xerror"]),"CP"])
plot(spam7b.tree,uniform=TRUE)
text(spam7b.tree,cex=0.75)
```

ADDENDA.

Se puede usar la opción de WEKA:

```
http://maya.cs.depaul.edu/~classes/ect584/WEKA/
http://maya.cs.depaul.edu/~classes/ect584/WEKA/classify.html

# Arboles con WEKA (a traves de R)
library(RWeka)
tree <- make_Weka_classifier("weka/classifiers/trees/J48",
c("bar","Weka_tree"))
print(tree)
WOW(tree)
```

```
fm <- tree(yesno~crl.tot+dollar+bang+money+n000+make, data=spam7,
control=Weka_control(U=TRUE,S=TRUE,M=150))
fm
table(observed = spam7$yesno, predicted = fitted(fm))
summary(fm)
plot(fm)
```