PRÁCTICA 4: ANÁLISIS DE COMPONENTES PRINCIPALES.
ANÁLISIS ESTADÍSTICO MULTIVARIANTE.
GRADO EN CIENCIA E INGENIERÍA DE DATOS.

Sumario: En esta práctica mostramos cómo resumir la información contenida en muchas variables aleatorias (relacionadas) en unas pocas variables denominadas componentes principales. Para ello necesitaremos disponer de una muestra de las variables en estudio. El Análisis de Componentes Principales (ACP o PCA por sus siglas en inglés) se podrá usar para estudiar (resumir) la información contenida en la muestra y las principales características de sus individuos. También se podrá usar para detectar grupos.

1. Estudio inicial de los datos.

Siempre que se aplique una técnica de análisis multivariante es conveniente hacer un análisis inicial de los datos. Los conjuntos de datos disponibles en R en el "paquete" *Datasets* se pueden ver mediante: `data( )`. Tecleando:

```
d<-LifeCycleSavings
```

incluiremos los datos de ese fichero en el objeto `d`. Tecleando `d` o `View(d)` podemos ver que el fichero contiene 5 variables medidas en 50 países diferentes. Los detalles sobre estos datos se pueden ver con: `help(LifeCycleSavings)` donde se indica que:

sr: incremento de los ahorros personales 1960-1970.
*pop*15: % población menor de 15 años.
*pop*75: % población mayor de 75.
dpi: ingresos per-capita.
ddpi: crecimiento del dpi 1960-1970.

Los valores de la primera columna se pueden obtener mediante `d$sr` o con `d[,1]`. Para resumir estas variables podemos hacer:

```
summary(d)
```

que proporciona el mínimo, el primer cuartil muestral, la mediana (segundo cuartil) muestral, la media, el tercer cuartil y el máximo. En este caso se aprecia que las variables tienen escalas muy diferentes. También podemos usar `str(d)` que nos dice las variables que hay, de qué tipo son, etc.

Para estudiar las correlaciones podemos hacer:

```
plot(d,pch=20,cex=0.8)
```

La gráfica se puede ver en la Figura 1, izquierda. Allí comprobamos que hay correlaciones positivas, negativas y variables independientes o con poca relación.

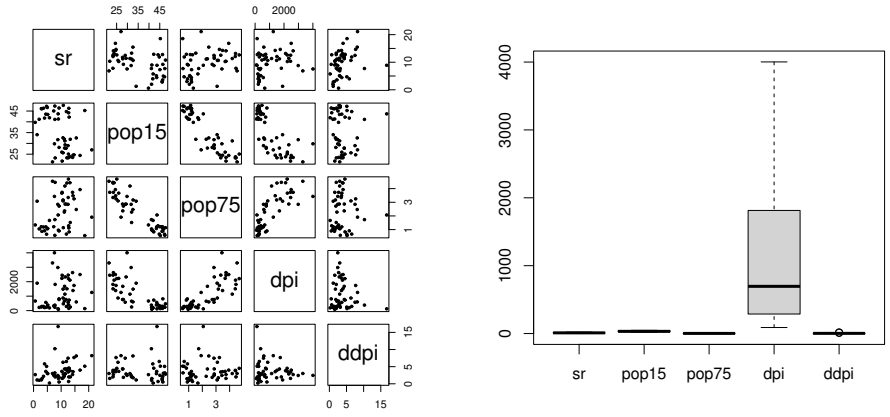

Figura 1: Gráficos bidimensionales para las 5 variables en estudio.

Para calcular el vector de medias no podemos usar `mean(d)`. Debemos usar `colMeans(d)` y si queremos guardarlo `mu<-colMeans(d)`. Otras opción es `sapply(d,mean)`.

Podemos calcular las matrices de covarianzas y correlaciones con `cov(d)` y `cor(d)`, respectivamente. De nuevo se aprecia que las escalas (varianzas) son muy diferentes y que existen variables con correlaciones (lineales) positivas, negativas y casi nulas. Estas relaciones se verán reflejadas en el PCA. Si queremos guardar la matriz de correlaciones en el objeto `M` haremos: `M<-cor(d)`. Tecleando `M` obtendremos:

Cor	sr	pop15	pop75	dpi	ddpi
sr	1.0000000	-0.45553809	0.31652112	0.2203589	0.30478716
pop15	-0.4555381	1.00000000	-0.90847871	-0.7561881	-0.04782569
pop75	0.3165211	-0.90847871	1.00000000	0.7869995	0.02532138
dpi	0.2203589	-0.75618810	0.78699951	1.0000000	-0.12948552
ddpi	0.3047872	-0.04782569	0.02532138	-0.1294855	1.00000000

Las escalas, la simetría y la posible existencia de valores atípicos (respecto de la normal) se pueden estudiar con los gráficos caja-bigote mediante: `boxplot(d)` o `boxplot(d[,i])` con $i = 1, \ldots, 5$ que incluyen la mediana, los cuartiles 1 y 3 (caja), y el mínimo y el máximo (bigotes), señalando los valores atípicos (con respecto a la normal) fuera de los "bigotes" (cuando existan). En el primero vemos las distintas escalas de las variables (especialmente dpi) por lo que es necesario representarlos individualmente. Un vez hecho esto en algunos de ellos se aprecia falta de simetría y en el último de estos gráficos que dos países (Libia y Jamaica) presentan valores de *ddpi* anormalmente grandes. Podemos usar los comandos `which.max(d[,5])`, `sort(d[,5])` u `order(d[,5])` para detectar los países con valores extremos (altos) en *ddpi*. También se puede usar `View(d)`. Aunque son valores atípicos para la distribución normal, estos datos son correctos y no se deben borrar.

Otra opción interesante son los histogramas que se pueden realizar con `hist(d$sr)` o con `hist(d[,1])`. Estos gráficos nos permiten estudiar simetrías y normalidad. Para añadir la curva normal a estos gráficos podemos hacer

```
hist(d$sr,probability=T,xlab='sr',ylab='f',main='Histograma sr')
curve(dnorm(x,mean(d$sr),sd(d$sr)),add=T,col='red')
```

El resultado se puede ver en la Figura 2, izquierda. Se aprecia una clara asimetría y falta de normalidad. Realice los gráficos para las otras variables.

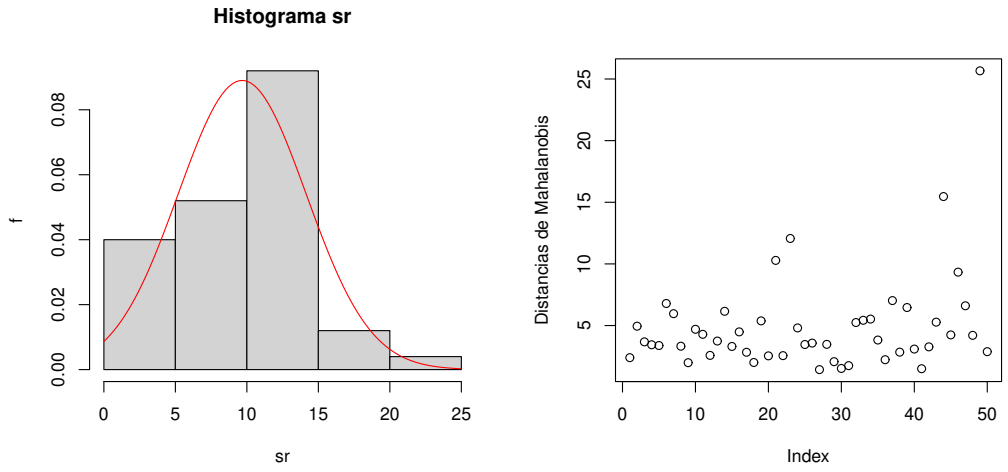

Figura 2: Histograma para la variable sr (izquierda) y distancias de Mahalanobis al cuadrado (derecha).

Las distancias de Mahalanobis a la media se pueden usar para detectar valores atípicos. En R se pueden calcular y representar con:

```
md<-mahalanobis(d,colMeans(d),cov(d))
md
plot(md,ylab='Distancias de Mahalanobis')
```

que proporciona las distancias al cuadrado y las representa. Los países mas "raros" (lejanos de la media) son USA (15.46) y Libia (25.66).

2. Cálculo de las componentes principales.

Para calcular las componentes principales debemos decidir en primer lugar si usamos la matriz de covarianzas o la de correlaciones. Con la primera opción, las variables no se cambian de escala y tendrán más importancia aquellas variables que tengan varianzas mayores. Con la segunda, las variables originales se estandarizan y todas tienen a priori la misma importancia (varianza uno). Esta última opción se usa cuando las variables se miden en unidades diferentes (al estandarizar, las unidades desaparecen) o tienen rangos (varianzas) muy diferentes. La primera opción se usa cuando las unidades son iguales y queremos mantener las escalas (la variables que varían poco tienen poca importancia).

Para nuestro conjunto de datos está claro que debemos usar la matriz de correlaciones ya que las escalas son muy diferentes (ver Figura 1, derecha). Posteriormente veremos qué ocurre si usamos la matriz de covarianzas.

En R existen dos comandos para calcular las componentes principales: `princomp` y `prcomp`. Como las componentes son únicas (salvo cambio de signo) los resultados serán muy similares pero puede haber pequeñas diferencias debidas a los métodos numéricos usados para su cálculo.

Para hacer un PCA con `princomp` usando la matriz de correlaciones de los datos en `d` basta teclear:

```
PCA<-princomp(d,cor=TRUE)
```

Para ver las características principales haremos:

```
summary(PCA,loadings=TRUE)
```

obteniendo:

Importance:	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5
Stan. dev.	1.6799041	1.1207437	0.777512	0.4895354	0.278721
Prop. Var.	0.5644156	0.2512133	0.120905	0.0479299	0.015537
Cum. Prop.	0.5644156	0.8156289	0.936534	0.984463	1

Loadings:	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5
sr	0.308	0.554	0.750	-0.130	-0.134
pop15	-0.571			-0.416	-0.707
pop75	0.560	-0.101	-0.212	0.390	-0.692
dpi	0.514	-0.266	-0.145	-0.801	
ddpi		0.782	-0.609	-0.123	

La importancia de las componentes se mide con sus desviaciones estándar (raíces cuadradas de los valores propios de la matriz de correlaciones ordenados de mayor a menor), la proporción en tanto por uno de sus varianzas y las proporciones acumuladas. En este caso, las varianzas iniciales suman 5 (la traza de la matriz de correlaciones) por lo que el primer valor de las varianzas $\widehat{\lambda}_1 = 2.822078$ se calcula con `1.6799041^2` y el de las proporciones 0.5644156 como `1.6799041^2/5`. Las proporciones acumuladas se calculan sumando las de las componentes anteriores. Por ejemplo,

$$0.8156289 = \frac{\widehat{\lambda}_1 + \widehat{\lambda}_2}{\widehat{\lambda}_1 + \widehat{\lambda}_2 + \widehat{\lambda}_3 + \widehat{\lambda}_4 + \widehat{\lambda}_5} = 0.5644156 + 0.2512133.$$

Estos valores nos indican que la primera componente mantiene un 56.44156 % de la información inicial, la segunda un 25.12133 % y las dos juntas un 81.56289 %.

Las cargas (*loadings*) son los vectores propios unitarios de los valores propios anteriores. Los valores ausentes son números pequeños (pero no necesariamente cero). Si queremos guardarlos podemos hacer:

```
PCA$loadings->L
```

De esta forma, tecleando `L[,1]` obtenemos el primer vector propio que se utiliza para calcular la primera componente principal. Su último coeficiente es 0.03787232 (pequeño pero no cero). Se puede comprobar que este vector tiene norma uno. Por lo tanto la primera componente principal se calcularía como:

$$Y_1 = 0.308 * X_1^* - 0.571 * X_2^* + 0.560 * X_3^* + 0.514 * X_4^* + 0.0379 * X_5^* \tag{1}$$

(los coeficientes se han redondeado), donde $X_i^* = (X_i - \bar{X}_i)/S_i$ es la variable i-ésima estandarizada (muestral) ya que usamos la media $\bar{X}_i$ y cuasi-desviación típica S_i muestrales de cada variable. Si usamos la matriz de covarianzas para el PCA, en esta fórmula se usarán las variables originales X_i.

Análogamente, las puntuaciones (*scores*), es decir, los valores que obtendrían los individuos de la muestra (países en este caso) en las componentes principales (usando esas fórmulas), se pueden calcular mediante:

```
PCA$scores->S
```

Tecleando `S[,1]` comprobamos que, por ejemplo, la puntuación en la primera componente de Australia es 1.36528994. El significado de estos valores se verá en la sección siguiente. Para comprobar que los valores obtenidos con `princomp` son los correctos podemos hacer: `eigen(M)` con lo que obtenemos los valores propios ordenados (varianzas de las componentes) y los vectores propios (cargas o coeficientes). Compruebe que coinciden con los valores obtenidos anteriormente.

Para comprobar los valores de las puntuaciones debemos primero estandarizar (por columnas) los datos iniciales (este paso no será necesario cuando usemos la matriz de covarianzas). Para ello usaremos: `z<-scale(d)` y, para calcular las puntuaciones de la primera componente, haremos:

```
y1<-0.30846174*z[,1]-0.57065322*z[,2]+0.56043119*z[,3]+0.51350640*z[,4]+0.03787232*z[,5]
```

De esta forma, para Australia obtenemos 1.35156808. La pequeña diferencia con el valor obtenido anteriormente en los scores se debe a que `princomp` usa varianzas y no cuasivarianzas para estandarizar. Es decir las puntuaciones anteriores (en $S[,1]$) se obtienen haciendo: `y1*sqrt(50/49)`. En general, se obtienen con $S[,i] = y_i\sqrt{n/(n-1)}$. Lógicamente, es mejor el calculo automático proporcionado por el comando `princomp`.

Las componentes principales se pueden calcular aunque no se dispongan de los datos completos usando únicamente la matriz de correlaciones (o covarianza). Para ello haremos: `princomp(covmat=M)` sustituyendo `M` por la matriz de correlación (o covarianzas) de los datos. Las cargas se calcularán como antes pero, en este caso, no podremos calcular las puntuaciones ya que no disponemos de datos en individuos.

Para calcular las componentes principales con `prcomp` usando la matriz de correlaciones debemos hacer: `PCAbis<-prcomp(d,scale=TRUE)`. Haciendo `summary(PCAbis)` se obtiene la importancia de las componentes, con `PCAbis$rotation` las cargas y con `PCAbis$x` las puntuaciones. Note que se obtienen las puntuaciones correctas (calculadas con las cuasivarianzas) pero que algunas aparecen cambiadas de signo (su interpretación será la opuesta respecto de lo obtenido con `princomp`).

Compruebe que se obtienen resultados totalmente diferentes (erróneos en este caso) si usamos la matriz de covarianzas tecleando `princomp(d)` (procure no alterar los "objetos" usados anteriormente ya que se usarán en las secciones siguientes).

3. Análisis de componentes principales (PCA).

Para analizar las componentes principales calculadas en la sección anterior primero debemos fijarnos en la importancia de cada una. Hablaremos posteriormente sobre el número adecuado de componentes pero, antes de analizarlas, debemos tener en cuenta que la primera tiene un 56.44156 % de la información inicial, la segunda un 25.12133 % y las dos juntas un 81.56289 %. Por lo tanto, en este caso, la información proporcionada por la primera será en general el doble de importante que la que proporciona la segunda, etc. Esto es un cómputo global por lo que puede haber variables que estén mejor representadas en Y_2 que en Y_1 (por ejemplo *ddpi*).

En segundo lugar miraremos las cargas (loadings) o coeficientes de las componentes que queremos analizar para poder dar un significado a estas variables nuevas denominadas componentes principales. Si miramos las cargas de Y_1 dadas en (1), teniendo en cuenta que las variables están estandarizadas (y tendrán valores similares), podemos afirmar que las variables que más influyen en Y_1 son (por orden de influencia): pop15 (negativa), pop75 (positiva), dpi (positiva) y sr (positiva). Por lo tanto, Y_1 tomará valores grandes en los países con valores pequeños en pop15 y grandes en las otras tres. Por lo tanto, Y_1 nos indicará los países que tienen poblaciones envejecidas (alta pop75 y baja pop15) y ricos (altos valores en dpi y sr). Estas suelen ser características de países muy

desarrollados económicamente.

Una vez que ya hemos interpretado una componente, podemos analizar sus puntuaciones para decir cómo serán (aproximadamente) los individuos de la muestra según los valores que toman en esa componente. Usando `summary(S[,1])`, `plot(S[,1],ylab='Y1')` (ver Figura 3, izquierda) o `sort(S[,1])` observamos que los valores de la muestra en Y_1 están entre -2.258755 y 2.787708 (su media es cero ya que hemos usado variables estandarizadas). Haciendo which.max(S[,1]) comprobamos que el valor mayor en Y_1 corresponde a Suecia (2.787708) y el menor a Malasia (-2.258755). Por lo tanto, Australia con 1.36528994 sería, en aquella época, un país bastante desarrollado y España con 0.69294913 estaría un poco por encima de la media. En esta gráfica se observa que casi no hay valores entre 0 y -1, por lo que la mayoría de los países se podrían clasificar como del tercer o del primer mundo (en esa época). Analice la segunda componente y estudie las puntuaciones de estos países en esa componente (ver Figura 3, derecha).

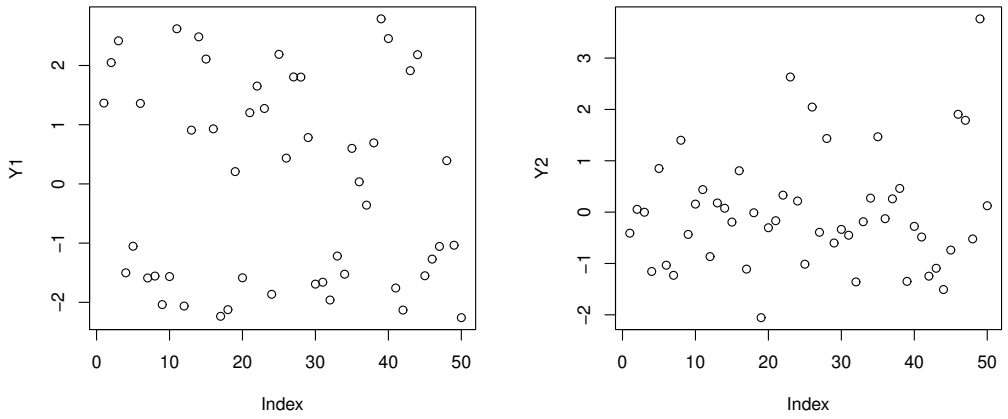

Figura 3: Gráfico de puntuaciones para las dos primeras componentes principales.

Como las componentes son incorreladas (independientes si las variables iniciales son normales), las podemos estudiar por separado. Sin embargo, muchas veces resulta conveniente representarlas por parejas en gráficos bidimensionales. Para representar las cargas y las puntuaciones estandarizadas de las dos primeras componentes haremos:

```
biplot(PCA,pc.biplot=TRUE)
```

El resultado puede verse en la Figura 4, izquierda. Las cargas aparecen como vectores en rojo con las escalas en la derecha y arriba y las puntuaciones estandarizadas con las etiquetas de los datos (nombres de los países) en negro con las escalas abajo (Y_1) y en la izquierda (Y_2).

Este gráfico es la 'mejor' proyección bidimensional de los ejes iniciales de las cinco variables y de las puntuaciones (estandarizadas) de los individuos (países) de la muestra. Las cargas de este gráfico se pueden usar (igual que antes) para interpretar las componentes. Las variables con vectores largos (norma cercana a 1) estarán bien representadas por las dos primeras componentes, mientras que las que tengan vectores cortos estarán mal representadas (se pierden al proyectar por ser casi perpendiculares). En este ejemplo, todas las variables están bien representadas en este gráfico. Además, se aprecia que `pop75`, `dpi` y, en menor medida, `sr`, hacen crecer la primera componente, mientras que `pop15` la hace disminuir y `ddpi` no influye en ella. Lógicamente, la interpretación es la misma que antes. También podemos observar que la segunda componente crece cuando crece `ddpi` (incremento ingresos per-capita) y, en menor medida, `sr` (incremento de los ahorros personales),

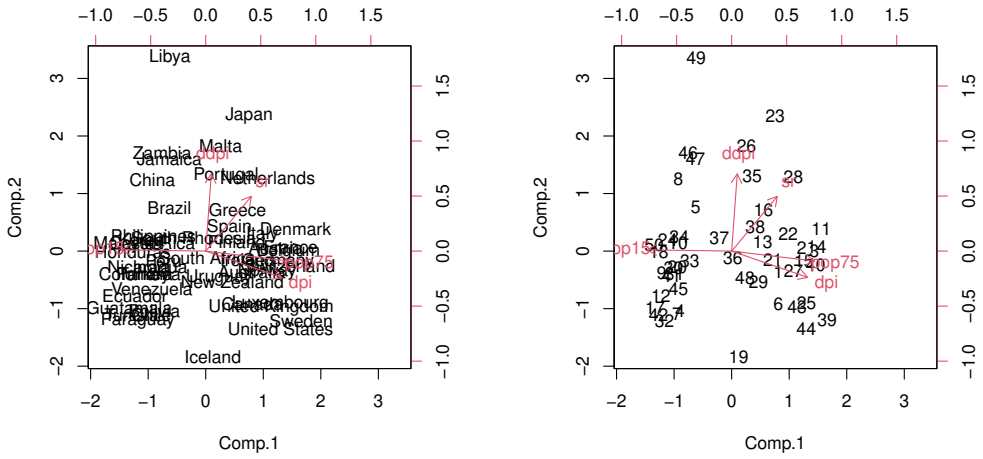

Figura 4: Gráfico de puntuaciones estandarizadas para las dos primeras componentes principales.

decrece un poco si crece `dpi` y casi no se ve afectada ni por `pop15` ni por `pop75`. Por lo tanto, se podría interpretar como un índice del crecimiento en esa década.

Las puntuaciones se usarán para decir cómo serán (aproximadamente) los individuos de la muestra (países) en esas características. A la derecha tendremos a los países más desarrollados y con poblaciones envejecidas (Suecia, US, etc.) a la izquierda lo contrario (Honduras, Guatemala, etc.), arriba a los países que más se desarrollaron durante esa época (Líbia, Japón, etc.) y debajo los que menos (Islandia, US, Paraguay, etc.). También nos podemos fijar en una variable concreta. Por ejemplo, con respecto a `sr` podríamos decir que los países con mayores incrementos de los ahorros personales (valores sr) deberían ser Japón, Malta y Holanda. Si vemos los datos de `sr` (con `d` y `sort(d[,1])`) podemos comprobar que efectivamente Japón es el que tiene un valor mayor (21.10) pero que el segundo es Zambia (18.56). Es lógico que al proyectar las variables originales, se pierda algo de la información contenida en ellas.

Como las etiquetas de los datos (nombres de los países) son muy grandes, algunos de ellos no se aprecian bien en el gráfico. Para sustituirlos por sus números de línea (ver Figura 3, derecha) podemos hacer:

```
biplot(PCA,pc.biplot=TRUE,xlabs=1:50)
```

Si queremos hacer un gráfico de las componentes tercera y cuarta haremos:

```
biplot(PCA,pc.biplot=TRUE,choices=c(3,4),xlabs=1:50)
```

También podemos hacer un gráfico solo de las puntuaciones (sin estandarizar) de las dos primeras componentes con:

```
plot(S[,1],S[,2],xlab='Y1',ylab='Y2')
```

Para encontrar un individuo (país) bastar mirar sus puntuaciones (scores) en S. Por ejemplo, encuentre España y diga cómo serán sus medidas según su posición en el gráfico. Para poner una etiqueta al dato $i = 38$ en ese gráfico haremos:

```
text(S[38,1]+0.4,S[38,2],labels='Esp')
```

Para que pongan las etiquetas (debajo) en todos los países haremos:

```
text(S[,1],S[,2]-0.2,labels=row.names(d),cex=0.6)
```

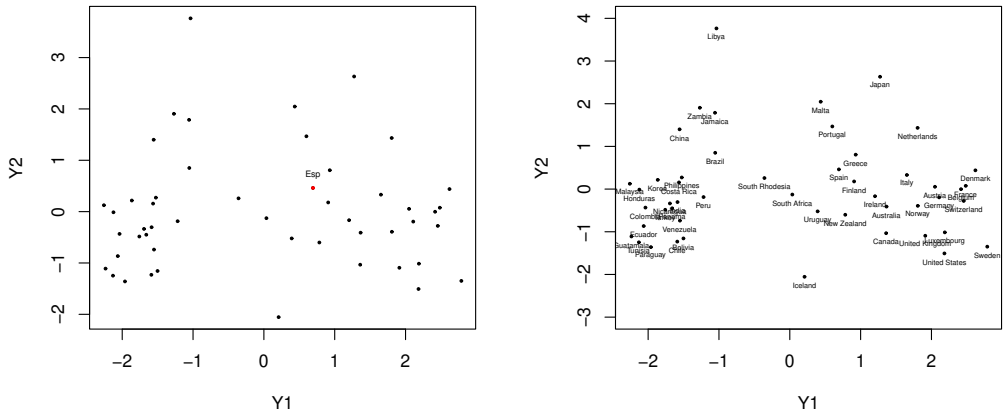

Figura 5: Gráfico de puntuaciones para las dos primeras componentes principales.

El resultado se puede ver en la Figura 5. Note que estos gráficos no coinciden con el realizado automáticamente por R ya que R dibuja las puntuaciones estandarizadas, es decir, hace:

```
plot(S[,1]/1.6799,S[,2]/1.1207)
```

En el gráfico $(S[,1], S[,2])$, la mayor dispersión de la primera componente nos indica que esta componente es más importante (tiene más información) a la hora de distinguir los datos. Las cargas se pueden representar de forma similar. Aunque no es muy habitual, las tres primeras componentes se podrían representar en gráficos 3D o mediante: `pairs(PCA$scores[,1:3])`.

4. Saturaciones.

Para medir las relaciones lineales entre las variables iniciales y las componentes principales, se puede calcular la matriz de correlaciones entre ellas conocida como matriz de saturaciones. Teóricamente, las correlaciones se calculan mediante:

$$a_{i,j} = Corr(X_i, Y_j) = \frac{t_{i,j}}{\sigma_i}\lambda_j^{1/2}.$$

En la práctica trabajaremos con sus estimaciones. Si el PCA se ha realizado con la matriz de correlaciones (o si todas las varianzas son 1) bastará con multiplicar la columna de los coeficientes (cargas) $t_{i,j}$ de cada componente principal por la raíz cuadrada de su valor propio (su desviación estándar) $\lambda_j^{1/2}$. Las saturaciones al cuadrado nos indicarán cuanta información (en tanto por 1) tendrá cada componente de cada variable. En nuestro ejemplo, las saturaciones de la primera componente principal se calcularán con:

```
S1<-L[,1]*1.6799041
```

Var.	sr	pop15	pop75	dpi	ddpi
Sat.	0.5181861	-0.9586427	0.9414706	0.8626415	0.0636219
Inf.	0.26851688	0.91899580	0.88636698	0.74415038	0.00404774

y las saturaciones al cuadrado (información) con `S1^2` obteniendo:

De esta forma, comprobamos que la variable mejor representada en Y_1 es *pop15* con un 91.89958 % y que de la última variable *ddpi* Y_1 prácticamente no tiene ninguna información (0.4047742 %).

Para calcular todas las saturaciones (usemos o no la matriz de correlaciones) podemos hacer: `SAT<-cor(d,S)` obteniendo:

Sat.	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5
sr	0.5181861	0.6211673	0.5832461	-0.0637125	-0.03740324
pop15	-0.9586427	0.0142035	0.0206425	-0.2037487	-0.19713655
pop75	0.9414706	-0.1131882	-0.1647000	0.1911276	-0.19278378
dpi	0.8626415	-0.2984564	-0.1127514	-0.3922864	0.01310945
ddpi	0.0636219	0.8764293	-0.4733754	-0.0604464	0.00927110

Como las componentes son incorreladas, las correlaciones múltiples al cuadrado o comunalidades serán la suma de las correlaciones al cuadrado:

$$Corr^2(X_i, (Y_1, \ldots, Y_p)) = \sum_{j=1}^{p} Corr^2(X_i, Y_j).$$

Estos valores nos indicarán la información (en tanto por 1) que mantienen las p primeras componentes sobre cada variable. Por ejemplo, si decidimos usar las dos primeras componentes, las correlaciones múltiples se calcularán mediante:

```
COM2<-SAT[,1]^2+ SAT[,2]^2
```

obteniendo:

Inf.	Comp.1	Comp.2	Comunalidad
sr	0.268516885	0.38584877	0.6543657
pop15	0.918995810	0.00020174	0.9191976
pop75	0.886366987	0.01281156	0.8991785
dpi	0.744150387	0.08907624	0.8332266
ddpi	0.004047742	0.76812824	0.7721760

Se observa que la variable mejor representada en las dos primeras componentes (gráfico *biplot*) es `pop15` de la que se mantiene un 91.91976 % de su información y que la peor representada es `sr` con un 65.43657 % (no es necesario usar tantos decimales). Se puede comprobar que la media de los valores de la última columna es 0.8156289 que coincide con la información que (en promedio) mantienen Y_1 y Y_2 (calculada anteriormente con `summary(PCA)`). Compruebe que ocurre lo mismo con las informaciones individuales de cada componente. También se puede comprobar que las sumas de los valores de cada columna nos dan los valores propios (informaciones) de cada componente principal (solo si usamos la matriz de correlaciones). Por ejemplo, compruebe que sumando los valores de la primera columna obtenemos 2.822078, es decir, el mayor valor propio de la matriz de correlaciones.

La correlación múltiple al cuadrado $Corr^2(X_i, (Y_1, \ldots, Y_p))$ es el máximo de las correlaciones que se pueden obtener con combinaciones lineales de las componentes $Y_1, \ldots, Y_p$. Además, el máximo de esas correlaciones se obtiene con los coeficientes incluidos en la matriz de cargas L. Por ejemplo, la mejor combinación lineal de las dos primeras componentes para estimar (linealmente) sr es la que se obtiene cortando `L[1,]`, es decir, $Z_1 = 0.3084617 * Y_1 + 0.5542456 * Y_2$. De esta forma, si calculamos Z_1 con:

```
Z1<-0.3084617*S[,1]+ 0.5542456*S[,2]
```

y calculamos `cor(d[,1],Z1)^2`, se obtiene 0.6543657 que coincide con la información que mantienen esas dos componentes sobre `sr`. La variable Z_1 se podría usar para predecir `sr` usando las técnicas de regresión lineal mediante

```
lm(d$sr~Z1)
plot(Z1,d$sr,pch=20,ylab='sr')
abline(lm(d$sr~Z1),col='red')
plot(d$sr-9.671-4.435*Z1,pch=20,ylab='Residuos')
```

Así observamos que la mejor manera de recuperar `sr` usando Z_1 es mediante $sr \approx 9.671 + 4.435 Z_1$. Las gráficas de la Figura 6 nos muestran los datos de ambas variables y los residuos (sus diferencias). Por ejemplo, para Australia obtendríamos la aproximación 10.530575 cuando su verdadero valor es 11.43. Para obtener mejores aproximaciones debemos aumentar el número de componentes m. Lógicamente, con $p = k = 5$ obtendremos los valores exactos.

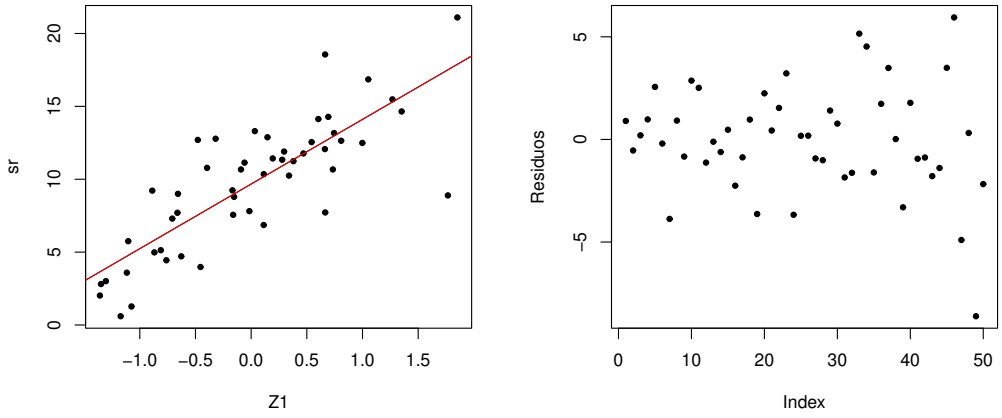

Figura 6: Gráfico de puntos y recta de regresión (rojo) para `sr` y Z_1 (izquierda) y residuos para la estimación con la recta de regresión (derecha).

5. Número de componentes.

Una vez realizado un PCA podemos preguntarnos con cuántas componentes principales debemos quedarnos. Las respuesta no es única y puede depender de factores subjetivos. Todas las soluciones serán correctas ya que lo que estamos haciendo es perder algo de información (la menor posible) a

cambio de reducir la dimensión inicial (número de variables). A continuación comentamos algunas de las técnicas más usadas. En todos ellas el número de componentes elegidas se representará por m y, lógicamente, siempre se tomarán las m primeras componentes principales (ya que son las que más información tienen).

5.1. Fijar un número concreto de componentes. Una opción válida es fijar un número de componentes concreto. Por ejemplo, si queremos hacer una única gráfica bidimensional, evidentemente, debemos tomar $m = 2$, con lo que únicamente analizaremos Y_1 e Y_2. En esta opción es fundamental incluir la información total mantenida por las componentes elegidas y advertir si ese número es bajo. Se suelen tomar números pares de componentes para poder realizar gráficas bidimensionales y el valor más usual es $m = 2$. Tecleando `summary(PCA)` (ver sección 2) comprobamos que, en nuestro ejemplo, si tomamos $m = 2$, mantendríamos un 81.56 % de la información inicial lo que podemos considerar como aceptable al reducir la dimensión de 5 a 2. También podríamos informar sobre las comunalidades, es decir, sobre la información mantenida por esas componentes de cada variable (ver sección anterior). En nuestro ejemplo, para $m = 2$, la variable peor representada es *sr* de la que mantienen un 65.44 %. Por lo tanto, todas las variables están bien representadas. En otros ejemplos nos podremos encontrar con variables que no están representadas en las componentes elegidas. En estos casos es importante señalarlo y, si fuera necesario, aumentar m.

5.2. Fijar un porcentaje mínimo de información mantenida. Si queremos mantener al menos un porcentaje p de la variabilidad inicial deberemos quedarnos con las primeras m componentes que verifiquen

$$\frac{\widehat{\lambda}_1 + ... + \widehat{\lambda}_m}{\widehat{\lambda}_1 + ... + \widehat{\lambda}_k} \geq \frac{p}{100},$$

donde $\widehat{\lambda}_i$ representan las estimaciones de los valores propios (ver sección 2). En nuestro ejemplo, si queremos mantener más de un $p = 90\,\%$, debemos tomar $m = 3$ con lo que mantendríamos un 93.65 %.

Otra regla (diferente) podría ser el fijar un porcentaje mínimo para las comunalidades. De esta forma nos aseguramos de que todas las variables originales (sean importantes o no), estén representadas en las componentes. En nuestro ejemplo, si queremos que las comunalidades sean mayores que 0.5 (es decir queremos mantener al menos un 50 % de todas las variables), debemos tomar $m = 2$. Nótese que con esta regla, en nuestro ejemplo, nunca obtendríamos $m = 1$ a pesar de que Y_1 tiene un 56 % de la información total.

5.3. Regla de Rao. Esta regla establece que solo serán relevantes las componentes que tengan una variabilidad (varianza o valor propio) mayor que la variabilidad mínima de las variables originales. De esta forma, se tiene

$$\text{máx}\, m : \widehat{\lambda}_m \geq \min_j \{S_j^2\},$$

donde S_j^2 representan a las cuasivarianzas muestrales de las variables originales. Si las componentes se calculan usando la matriz de correlaciones, como esto es equivalente a usar las variables estandarizadas, se entiende que las varianzas son 1 y, por lo tanto, se toman solo las componentes con valores propios (varianzas o desviaciones típicas) mayores que uno. En nuestro ejemplo, esta regla nos conduce a $m = 2$. Si calculásemos las componentes con la matriz de covarianzas (aunque ya hemos comentado que esto no sería correcto en este ejemplo), el mínimo de las cuasivarianzas muestrales corresponde a la variable `pop75` y vale 1.66609082 (hacer `var(d$pop75)` o `cov(d)`) y los valores propios de la matriz de covarianzas valen: 981871.2, 43.14338, 13.68328, 6.629537 y 0.2351568 por lo que, con este criterio, tomaríamos $m = 4$.

5.4. Regla de Kaiser. Esta regla es similar a la anterior y establece que solo serán relevantes las componentes que tengan una variabilidad mayor que la variabilidad media de las variables originales. De esta forma, se tiene

$$\text{máx}\, m : \widehat{\lambda}_m \geq \frac{1}{k}\sum_{j=1}^{k} S_j^2.$$

Si usamos la matriz de correlaciones para calcular las componentes, como las varianzas iniciales son 1, su media es 1 y este criterio coincide con el de Rao por lo que, en nuestro ejemplo, obtenemos el mismo resultado $m = 2$. Si calculásemos las componentes con la matriz de covarianzas (aunque no sea correcto), la media de las cuasivarianzas muestrales es 196387 y los valores propios de la matriz de covarianzas valen: 981871.2, 43.14338, 13.68328, 6.629537 y 0.2351568 por lo que, con este criterio, tomaríamos $m = 1$.

5.5 Regla del codo o del gráfico de sedimentación. Es uno de los métodos más usados y suele ir incluido en casi todos los programas de estadística. El método consiste en representar j (eje x) frente a los valores propios estimados $\widehat{\lambda}_j$ obteniéndose el denominado gráfico de sedimentación o desmoronamiento (*scree graph*). El gráfico será similar a la acumulación de sedimentos en la ladera de una montaña (cono de desmoronamiento). Se trataría de separar "la montaña" de los "sedimentos". La regla establece que serán representativas las componentes hasta el primer "codo" (sin incluirlo) de la gráfica o hasta que comience la línea recta aproximada final (separando los sedimentos de la montaña). Para realizar este gráfico de forma automática en R haremos: `screeplot(PCA)` con lo que se obtiene el gráfico de la Figura 7 (izquierda). Se puede obtener un gráfico similar (derecha) mediante:

```
plot(eigen(cor(d))$values,type='l',ylab='valores propios')
```

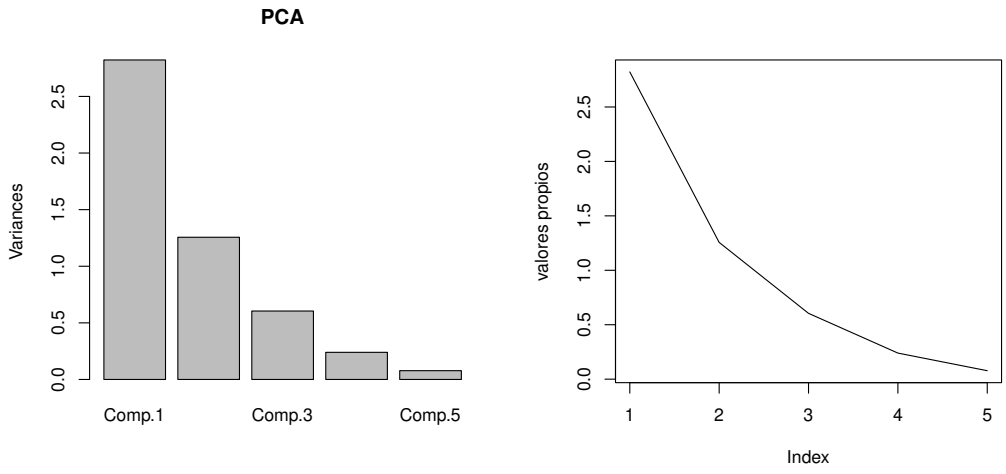

Figura 7: Gráfico del codo (scree graph).

En estos gráficos, aunque no está muy claro, parece que el codo (los sedimentos) se encuentra en $j = 3$, por lo que tomaríamos las dos primeras componentes ($m = 2$). Las soluciones $m = 1$ y $m = 3$ también serían aceptables. En otras ocasiones el "codo" aparece más claro y solo hay una opción (especialmente cuando k es más grande).

5.6 Prueba de esfericidad. Esta regla se basa en la contrastación de la hipótesis

$$H_0 : \lambda_{m+1} = ... = \lambda_k,$$

frente a su contraria (alguno es diferente) cuyo significado es que para un m dado, las componentes restantes podrían tener igual variabilidad teórica (las diferencias en las varianzas muestrales se deben al azar) y, por lo tanto, no debemos dar preferencia a una sobre otra. Esto esd equivalente a que haya "esfericidad" en el vector $(Y_{m+1}, \ldots, Y_k)$, es decir, que su matriz de covarianzas es proporcional a la matriz identidad (recuerde que sabemos que las componentes son incorreladas). El test de esfericidad de Bartlett se basa en el test de razón de verosimilitudes que (bajo normalidad) da el estadístico:

$$T = \left(n - \frac{2k+11}{6}\right)(k-m)\ln\left(\frac{ma}{mg}\right),$$

donde $ma = \frac{1}{k-m}\sum_{i=m+1}^{k}\widehat{\lambda}_i$ y $mg = (\prod_{i=m+1}^{k}\widehat{\lambda}_i)^{1/(k-m)}$ son las medias aritmética y geométrica de los últimos valores propios. En condiciones de normalidad de los datos iniciales (que en nuestros caso no se verificaban) y bajo H_0, T sigue una distribución chi-cuadrado con $gl = 0.5(k-m-1)(k-m+2)$ grados de libertad. Si H_0 no es cierta, T tiende a tomar valores mayores por lo que la región de rechazo sería de la forma $T > \chi^2_{1-\alpha,gl}$, donde $\chi^2_{1-\alpha,gl}$ es el cuantil $1-\alpha$ de esa chi-cuadrado.

Para aplicar este test a nuestro ejemplo con $m = 2$ calcularemos

```
eigen(cor(d))$values->Lambda
mean(Lambda[3:5])->ma
exp(mean(log(Lambda[3:5])))->mg
(50-(2*5+11)/6)*(5-2)*log(ma/mg)->TB
0.5*(5-2-1)*(5-2+2)->gl
1-pchisq(TB,gl)
```

obteniendo:

$$\begin{aligned}
m_a &= \frac{1}{k-m}\sum_{i=m+1}^{k}\widehat{\lambda}_i = 0.3072852\\
m_g &= \left(\prod_{i=m+1}^{k}\widehat{\lambda}_i\right)^{1/(k-m)} = 0.2240993\\
T &= \left(n - \frac{2k+11}{6}\right)(k-m)\ln\left(\frac{m_a}{m_g}\right) = 44.03837\\
gl &= 0.5(k-m-1)(k-m+2) = 5\\
P-valor &= \Pr(\chi^2_5 > 44.03837) = 2.27505 * 10^{-8}
\end{aligned}$$

y, como el P-valor obtenido es muy pequeño (menor que 0.05), rechazaremos la esfericidad de las tres últimas componentes (H_0) por lo que, si queremos, podemos calcular más componentes y éstas no serán al azar. La región crítica (de rechazo) para este test con $\alpha = 0.05$ es $(11.0705, \infty)$ donde $q_{0.95} = 11.0705 = \chi^2_{0.95,5}$ se calcula mediante `qchisq(0.95,5)`. La gráfica de la función de densidad de una χ^2_5 se puede obtener con: `curve(dchisq(x,5),0,50)` (ver Figura 8).

Note que es muy posible que no haya esfericidad para ningún m ($m = 1, 2, 3$) (los valores propios teóricos son todos diferentes), pero esto no implica que tengamos que tomar todas las componentes principales. Si para algún m se acepta la esfericidad, no sería conveniente aumentar las componentes (ya que éstas podrían obtenerse por azar) y sí podríamos intentar disminuir m.

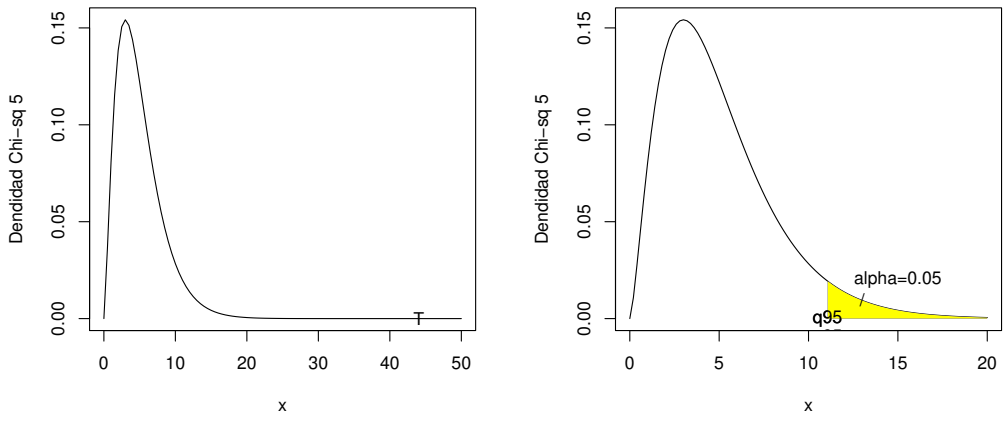

Figura 8: P-valor (izquierda) y región crítica (derecha) para el test de esfericidad.

6. Ejercicios.

1. Aplicar un PCA a los datos del fichero: `USArrests` incluido en R que contiene datos sobre los arrestos por cada 100000 residentes por asesinato, asalto o violación en cada uno de los 50 estados de USA en 1973. También se incluye el porcentaje de población que vive en las áreas urbanas. Fuente: `help(USArrests)`.

2. Aplicar un PCA a los datos de las columnas 5-10 del objeto `d` del fichero `bears.rda`[1] (aula virtual). Esas columnas contienen diversas medidas de 143 osos (Head.L= longitud de la cabeza (pulgadas), Head.W=anchura de la cabeza (pulgadas), Neck.G=perímetro cuello (pulgadas), Length=altura (pulgadas), Chest.G=perímetro pecho (pulgadas), Weight=peso (libras)). Incluir la variable `Sex` en los gráficos (esta variable no se puede incluir en el cálculo de las componentes principales)con:
`biplot(pca,pc.biplot=TRUE,xlabs=d$Sex)` y analizarlos. Fuente: Minitab15.

3. Aplicar un PCA a los datos del fichero `heptathlon` del paquete `MVA`[2] correspondientes a los resultados en la prueba femenina de heptatlon en las olimpiadas de Seul 1988.

4. Aplicar un PCA a los datos del fichero `pottery` del paquete `MVA`[2] que contiene resultados de análisis químicos de cerámica británica de la época romana de diversas regiones y hornos (kiln). La región 1 corresponde al horno 1, la región 2 a los hornos 2 y 3, y la región 3 a los hornos 4 y 5. ¿Podemos usar estas medidas para determinar el origen de la cerámica? Indicación: para incluir las regiones en los gráficos deberemos usar `biplot(pca,pc.biplot=TRUE,xlabs=d$kiln)`

5. Aplicar un PCA a los datos del objeto fichero `d` del fichero fichero `nota.rda`[1] (aula virtual) que contiene las notas (sobre 100) de alumnos de matemáticas en una universidad americana. Fuente: Rencher (1995, Methods of Multivariate Analysis, Wiley).

6. Aplicar un PCA a los datos del objeto fichero `d` del fichero fichero `madres.rda`[1] (aula virtual) que contiene las medidas de madres y sus bebés recién nacidos. Las variables son: PESOM

[1]Para leer este tipo de archivos teclear `load('f:/name.rda')` indicando la ruta completa en donde se encuentra el archivo y reemplazando name por el nombre del archivo.

[2]Para leer este conjunto de datos hay que instalar el paquete `MVA` pinchando en el menú: `Paquete >Instalar Paquete` seleccionando `MVA` y tecleando en R: `library('MVA')` (o indicando en el menú que se carge este paquete).

(peso madre), TALLAM (altura de la madre), SEM (semanas de gestación), PASM (presión sanguínea sistólica de la madre), PADM (presión sanguínea diastólica de la madre), PESOR (peso del recién nacido), TALLAR (altura recién del nacido), PTR (perímetro torácico del recién nacido), PCR (perímetro craneal del recién nacido).

7. Aplicar un PCA a los datos del objeto `d` del fichero fichero `decatlon.rda`[1] (aula virtual) que contiene los resultados obtenidos por 24 atletas en las 10 pruebas de décatlon en los Juegos Olímpicos de Seul 1988. Las variables corresponden a las pruebas siguientes: X1=100 metros lisos (en segundos), X2=salto de longitud (metros), X3=lanzamiento de peso (m), X4=salto de altura (m.), X5=400 metros lisos (seg.), X6=110 metros vallas (seg.), X7=lanzamiento de disco (m.), X8=salto con pértiga (m.), X9=lanzamiento de jabalina (m.), X10=1500 metros lisos (seg.) y X11=puntuación.