Lección 4

Introducción a la estadística descriptiva multidimensional

En general, los datos que se recogen en experimentos son multidimensionales: medimos varias variables aleatorias sobre una misma muestra de individuos, y organizamos esta información en tablas de datos en las que las filas representan los individuos observados y cada columna corresponde a una variable diferente. En las lecciones finales del primer volumen ya aparecieron datos cualitativos y ordinales multidimensionales, para los que calculamos y representamos gráficamente sus frecuencias globales y marginales; en esta lección estudiamos algunos estadísticos específicos para resumir y representar la relación existente entre diversas variables cuantitativas.

4.1. Vectores aleatorios

Vector aleatorio. Una *variable aleatoria p-dimensional*, o *vector aleatorio de dimensión* p, es un vector (fila) compuesto por p variables aleatorias

$$\underline{X} = (X_1, X_2, \ldots, X_p).$$

Como en el caso de las variables aleatorias unidimensionales, es importante distinguir entre los vectores aleatorios (los modelos teóricos), y las realizaciones o las muestras de los mismos, que corresponden a una o varias mediciones concretas de las variables que forman dichos vectores.

Por ejemplo, si llamamos X_1 a la variable aleatoria que da la edad de un individuo (en años), X_2 a la que da su altura (redondeada a cm) y X_3 a la que da su peso (redondeada a kg con una cifra decimal), entonces

$$\underline{X} = (X_1, X_2, X_3)$$

es un vector aleatorio de dimensión 3. Cada vez que medimos la edad, la altura y el peso de una persona, y organizamos estos datos en este orden como un vector numérico, obtenemos una realización de $\underline{X}$.

Sea ahora $\underline{X} = (X_1, X_2, \ldots, X_p)$ un vector aleatorio y, para cada $i = 1, \ldots, p$, sean μ_i y σ_i la media y la desviación típica, respectivamente, de su componente X_i. Entonces:

- El *valor esperado*, o *vector de medias*, de $\underline{X}$ es el vector formado por los valores esperados, o medias, de sus componentes:

$$E(\underline{X}) = (E(X_1), \ldots, E(X_p)) = (\mu_1, \ldots, \mu_p).$$

  También lo denotaremos simplemente $\boldsymbol{\mu}$.

- El *vector de varianzas* de $\underline{X}$ es el vector formado por las varianzas de sus componentes:

$$Var(\underline{X}) = (Var(X_1), \ldots, Var(X_p)) = (\sigma_1^2, \ldots, \sigma_p^2).$$

- El *vector de desviaciones típicas* de $\underline{X}$ es el vector formado por las desviaciones típicas de sus componentes:

$$\sigma(\underline{X}) = (\sigma(X_1), \ldots, \sigma(X_p)) = (\sigma_1, \ldots, \sigma_p).$$

Tipificación. Sea X una variable aleatoria de media μ y desviación típica σ. Recordemos que si $a, b \in \mathbb{R}$, entonces $aX + b$ es una variable aleatoria de media, varianza y desviación típica, respectivamente,

$$E(aX + b) = a\mu + b, \quad Var(aX + b) = a^2\sigma^2, \quad \sigma(aX + b) = |a|\sigma.$$

Llamaremos la *variable tipificada* de X a la variable aleatoria

$$Z = \frac{X - \mu}{\sigma}.$$

Las fórmulas anteriores implican que si Z es una variable tipificada, entonces $E(Z) = 0$ y $\sigma(Z) = 1$. Por ejemplo, cuando construimos una variable aleatoria normal estándar a partir de una variable normal X restándole su media y dividiendo el resultado por su desviación típica, lo que hacemos es tipificarla.

Si $\underline{X} = (X_1, \ldots, X_p)$ es un vector aleatorio, su *vector tipificado* $\underline{Z}$ se obtiene substituyendo cada X_i por su variable tipificada:

$$\underline{Z} = \Big(\frac{X_1 - \mu_1}{\sigma_1}, \ldots, \frac{X_p - \mu_p}{\sigma_p}\Big).$$

Covarianzas. Dadas dos variables aleatorias X_1 y X_2 de medias μ_1 y μ_2, respectivamente, su *covarianza* es

$$Cov(X_1, X_2) = E((X_1 - \mu_1) \cdot (X_2 - \mu_2)).$$

Es fácil comprobar que la covarianza también se puede calcular mediante la identidad

$$Cov(X_1, X_2) = E(X_1 \cdot X_2) - \mu_1 \cdot \mu_2.$$

En efecto

$$\begin{aligned} Cov(X_1, X_2) &= E((X_1 - \mu_1)(X_2 - \mu_2)) = E(X_1X_2 - \mu_1X_2 - \mu_2X_1 + \mu_1\mu_2) \\ &= E(X_1X_2) - \mu_1E(X_2) - \mu_2E(X_1) + \mu_1\mu_2 \\ &= E(X_1X_2) - \mu_1\mu_2 - \mu_2\mu_1 + \mu_1\mu_2 = E(X_1X_2) - \mu_1\mu_2 \end{aligned}$$

La covarianza de X_1 y X_2 puede tomar cualquier valor real, y mide el grado de variación conjunta de las variables. Si valores grandes de una variable corresponden a valores grandes de la otra, su covarianza es positiva. En el caso opuesto, si valores grandes de una variable corresponden a valores pequeños de la otra, su covarianza es negativa. Por lo tanto, el signo de la covarianza refleja la tendencia de la relación entre las variables. En cambio, su magnitud en valor absoluto no tiene una interpretación sencilla.

Si X_1 y X_2 son variables independientes, entonces su covarianza es 0, puesto que en este caso $E(X_1 \cdot X_2) = E(X_1) \cdot E(X_2) = \mu_1\mu_2$. El recíproco es falso: dos variables aleatorias pueden tener covarianza 0 y no ser independientes.

La covarianza es simétrica, $Cov(X_1, X_2) = Cov(X_2, X_1)$, y la covarianza de una variable aleatoria consigo misma es su varianza:

$$Cov(X, X) = E((X - \mu)^2) = Var(X).$$

Para simplificar la notación, se suele utilizar σ_{ij} para denotar la covarianza de dos variables aleatorias X_i y X_j que formen parte de un vector aleatorio. Es decir, escribiremos

$$\sigma_{ij} = Cov(X_i, X_j) \text{ y } \sigma_{ii} = Cov(X_i, X_i) = \sigma_i^2.$$

Igual que en el caso unidimensional, un vector aleatorio $\underline{X} = (X_1, \ldots, X_p)$ posee una medida de su dispersión respecto de su valor esperado $\boldsymbol{\mu}$. Es su *matriz de covarianzas*, que se define como

$$\begin{aligned} Cov(\underline{X}) = E((\underline{X} - \boldsymbol{\mu})^t \cdot (\underline{X} - \boldsymbol{\mu})) &= E\left(\begin{pmatrix} X_1 - \mu_1 \\ X_2 - \mu_2 \\ \vdots \\ X_p - \mu_p \end{pmatrix} \cdot (X_1 - \mu_1, X_2 - \mu_2, \ldots, X_p - \mu_p)\right) \\ &= \begin{pmatrix} E((X_1 - \mu_1)^2) & E((X_1 - \mu_1)(X_2 - \mu_2)) & \ldots & E((X_1 - \mu_1)(X_p - \mu_p)) \\ E((X_2 - \mu_2)(X_1 - \mu_1)) & E((X_2 - \mu_2)^2) & \ldots & E((X_2 - \mu_2)(X_p - \mu_p)) \\ \vdots & \vdots & \ddots & \vdots \\ E((X_p - \mu_p)(X_1 - \mu_1)) & E((X_p - \mu_p)(X_2 - \mu_2)) & \ldots & E((X_p - \mu_p)^2) \end{pmatrix} \\ &= \begin{pmatrix} \sigma_{11} & \sigma_{12} & \ldots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \ldots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \ldots & \sigma_{pp} \end{pmatrix} \end{aligned}$$

Es decir, la matriz de covarianzas de $\underline{X}$ tiene como entrada (i,j) la covarianza σ_{ij} de X_i y X_j. Se puede comprobar fácilmente que esta matriz se puede calcular mediante la identidad

$$Cov(\underline{X}) = E(\underline{X}^t \cdot \underline{X}) - \boldsymbol{\mu}^t \cdot \boldsymbol{\mu}.$$

La matriz de covarianzas de $\underline{X}$ también se suele representar por Σ.

Correlaciones. Como el valor concreto de la covarianza es difícil de interpretar, para medir la relación lineal entre dos variables aleatorias se usa el llamado *coeficiente de correlación lineal de Pearson* (o *correlación* a secas), que viene a ser una versión normalizada de la covarianza. En concreto, la *correlación* de las variables X_i y X_j se define como el cociente

$$Cor(X_i, X_j) = \frac{\sigma_{ij}}{\sigma_i \sigma_j},$$

y es una medida adimensional de la relación entre X_i y X_j. A menudo denotaremos $Cor(X_i, X_j)$ por medio de ρ_{ij}.

Las correlaciones tienen las propiedades siguientes:

(a) $-1 \leqslant \rho_{ij} \leqslant 1$.

(b) $\rho_{ij} = \rho_{ji}$ y $\rho_{ii} = 1$.

(c) Si $\sigma_i = \sigma_j = 1$, entonces $\rho_{ij} = \sigma_{ij}$.

(d) Si $a_i, a_j, b_i, b_j \in \mathbb{R}$ y $a_i, a_j \neq 0$, entonces

$$Cor(a_i X_i + b_i, a_j X_j + b_j) = \pm Cor(X_i, X_j),$$

donde el signo que aparece es el del cociente a_i/a_j.

(e) Si $\rho_{ij} = \pm 1$, las variables tienen una relación lineal perfecta, es decir, existen $\alpha, \beta \in \mathbb{R}$ tales que $X_i = \alpha X_j + \beta$. La pendiente α de esta recta tiene el mismo signo que la correlación.

(f) Si $\rho_{ij} = 0$, decimos que las variables X_i y X_j son *incorreladas*. Notemos que la correlación es 0 si, y sólo si, la covarianza es 0. Por lo tanto, dos variables aleatorias independientes son incorreladas. El recíproco en general es falso.

(g) La correlación de dos variables es igual a la covarianza de sus variables tipificadas,

$$Cor(X_i, X_j) = Cov\Big(\frac{X_i - \mu_i}{\sigma_i}, \frac{X_j - \mu_j}{\sigma_j}\Big).$$

La *matriz de correlaciones* de un vector aleatorio $\underline{X} = (X_1, \ldots, X_p)$ es

$$Cor(\underline{X}) = \begin{pmatrix} 1 & \rho_{12} & \ldots & \rho_{1p} \\ \rho_{21} & 1 & \ldots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \ldots & 1 \end{pmatrix}.$$

Por la propiedad (g) anterior, la matriz de correlaciones de un vector aleatorio $\underline{X}$ es igual a la matriz de covarianzas de su vector tipificado $\underline{Z}$.

4.2. Matrices de datos cuantitativos

Supongamos que hemos medido los valores de p variables aleatorias $X_1, \ldots, X_p$ sobre un conjunto de n individuos u objetos. Es decir, tenemos n observaciones de p variables. En cada observación, los valores que toman estas variables forman un vector que será una realización del vector aleatorio $\underline{X} = (X_1, X_2, \ldots, X_p)$. Para trabajar con estas observaciones, las dispondremos en una tabla de datos donde cada fila corresponde a un individuo y cada columna, a una variable. En `R`, lo más conveniente es definir esta tabla en forma de *data frame*, pero, por conveniencia de lenguaje, en el texto de esta lección la representaremos como una matriz

$$X = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}.$$

Utilizaremos las notaciones siguientes:

- Denotaremos la i-ésima fila de X por

  $$x_{i\bullet} = (x_{i1}, x_{i2}, \ldots, x_{ip}).$$

  Este vector está compuesto por las observaciones de las p variables sobre el i-ésimo individuo.

- Denotaremos la j-ésima columna de X por

  $$x_{\bullet j} = \begin{pmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{nj} \end{pmatrix}.$$

  Esta columna está formada por todos los valores de la j-ésima variable.

Observad que, en cada caso, el punto en el subíndice representa el índice «variable» de los elementos del vector o de la columna.

De esta manera, podremos expresar la matriz de datos X tanto por filas como por columnas:

$$X = \begin{pmatrix} x_{1\bullet} \\ x_{2\bullet} \\ \vdots \\ x_{n\bullet} \end{pmatrix} = (x_{\bullet 1}, x_{\bullet 2}, \ldots, x_{\bullet p}).$$

Con estas notaciones, podemos generalizar al caso multidimensional los estadísticos de una variable cuantitativa, definiéndolos como los vectores que se obtienen aplicando el estadístico concreto a cada columna de la tabla de datos. Así:

- El *vector de medias* de X es el vector formado por las medias aritméticas de sus columnas:
$$\overline{X} = (\overline{x}_{\bullet 1}, \overline{x}_{\bullet 2}, \ldots, \overline{x}_{\bullet p}),$$
donde, para cada $j = 1, \ldots, p$,
$$\overline{x}_{\bullet j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij}.$$
Observemos que
$$\begin{aligned}\overline{x} &= (\overline{x}_{\bullet 1}, \overline{x}_{\bullet 2}, \ldots, \overline{x}_{\bullet p}) = \frac{1}{n}\Big(\sum_{i=1}^{n} x_{i1}, \sum_{i=1}^{n} x_{i2}, \ldots, \sum_{i=1}^{n} x_{ip}\Big) \\ &= \frac{1}{n}\sum_{i=1}^{n}(x_{i1}, x_{i2}, \ldots, x_{ip}) = \frac{1}{n}\sum_{i=1}^{n} x_{i\bullet}\end{aligned}$$
Es decir, el *vector de medias* de X es la media aritmética de sus vectores fila.

- El *vector de varianzas* de X es el vector formado por las varianzas de sus columnas:
$$s_X^2 = (s_1^2, s_2^2, \ldots, s_p^2),$$
donde
$$s_j^2 = \frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{x}_{\bullet j})^2.$$

- El *vector de varianzas muestrales* de X está formado por las varianzas muestrales de sus columnas:
$$\widetilde{s}_X^2 = (\widetilde{s}_1^2, \widetilde{s}_2^2, \ldots, \widetilde{s}_p^2),$$
donde
$$\widetilde{s}_j^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \overline{x}_{\bullet j})^2 = \frac{n}{n-1}s_j^2.$$

- Los *vectores de desviaciones típicas* s_X y *de desviaciones típicas muestrales* $\widetilde{s}_X$ de X son los formados por las desviaciones típicas y las desviaciones típicas muestrales de sus columnas, respectivamente:
$$\begin{aligned}s_X &= (s_1, s_2, \ldots, s_p) = (\sqrt{s_1^2}, \sqrt{s_2^2}, \ldots, \sqrt{s_p^2}) \\ \widetilde{s}_X &= (\widetilde{s}_1, \widetilde{s}_2, \ldots, \widetilde{s}_p) = (\sqrt{\widetilde{s}_1^2}, \sqrt{\widetilde{s}_2^2}, \ldots, \sqrt{\widetilde{s}_p^2})\end{aligned}$$

Como en el caso unidimensional, $\overline{X}$ es un estimador de $E(\underline{X}) = \boldsymbol{\mu}$. Tanto s_X^2 como $\widetilde{s}_X^2$ son estimadores del vector de varianzas de $\underline{X}$: cuando las distribuciones de todas las variables aleatorias del vector son normales, el primero es el máximo verosímil y el segundo es insesgado.

Estos vectores de estadísticos se pueden calcular con `R` aplicando la función correspondiente al estadístico a todas las columnas de la tabla de datos. La manera más sencilla de hacerlo en un solo paso es usando la función `sapply`, si tenemos guardada la tabla como un *data frame*, o `apply` con `MARGIN=2`, si la tenemos guardada en forma de matriz.

Ejemplo 4.1. Consideremos la tabla de datos

$$X = \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}$$

formada por 4 observaciones de 3 variables; por lo tanto, $n = 4$ y $p = 3$. Vamos a guardarla en un *data frame* y a calcular sus estadísticos.

```
> X=data.frame(V1=c(1,1,2,3),V2=c(-1,0,3,0),V3=c(3,3,0,1))
> X
  V1 V2 V3
1  1 -1  3
2  1  0  3
3  2  3  0
4  3  0  1
> sapply(X, mean) #Vector de medias
  V1   V2   V3
1.75 0.50 1.75
> sapply(X, var) #Vector de varianzas muestrales
       V1        V2        V3
0.9166667 3.0000000 2.2500000
> sapply(X, sd) #Vector de desviaciones típicas muestrales
       V1        V2        V3
0.9574271 1.7320508 1.5000000
> var_ver=function(x){var(x)*(length(x)-1)/length(x)} #Varianza
   "verdadera"
> sd_ver=function(x){sqrt(var_ver(x))} #Desv. típica "verdadera"
> sapply(X, var_ver) #Vector de varianzas "verdaderas"
    V1     V2     V3
0.6875 2.2500 1.6875
> sapply(X, sd_ver) #Vector de desviaciones típicas "verdaderas"
       V1        V2        V3
0.8291562 1.5000000 1.2990381
```

Observación. De ahora en adelante, supondremos que todas las variables cuantitativas que aparezcan en lo que queda de lección, incluidas las columnas de tablas de datos, son no constantes y, por lo tanto, tienen desviación típica no nula.

4.3. Transformaciones lineales

A veces es conveniente aplicar una transformación lineal a una tabla de datos X, sumando a cada columna un valor y luego multiplicando cada columna resultante por otro valor. Los dos ejemplos más comunes de trasformación lineal son el *centrado* y la *tipificación* de datos.

Para *centrar* una matriz de datos X, se resta a cada columna su media aritmética:

$$\widetilde{X} = \begin{pmatrix} x_{11} - \overline{x}_{\bullet 1} & x_{12} - \overline{x}_{\bullet 2} & \dots & x_{1p} - \overline{x}_{\bullet p} \\ x_{21} - \overline{x}_{\bullet 1} & x_{22} - \overline{x}_{\bullet 2} & \dots & x_{2p} - \overline{x}_{\bullet p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} - \overline{x}_{\bullet 1} & x_{n2} - \overline{x}_{\bullet 2} & \dots & x_{np} - \overline{x}_{\bullet p} \end{pmatrix}.$$

Llamaremos a esta matriz la *matriz de datos centrados* de X.

Ejemplo 4.2. Consideremos de nuevo la matriz de datos del Ejemplo 4.1,

$$X = \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}.$$

Para centrarla, hemos de restar a cada columna su media:

$$\overline{x}_{\bullet 1} = \frac{1+1+2+3}{4} = 1.75,\ \overline{x}_{\bullet 2} = \frac{-1+0+3+0}{4} = 0.5,\ \overline{x}_{\bullet 3} = \frac{3+3+0+1}{4} = 1.75.$$

Por lo tanto, su matriz de datos centrados es

$$\widetilde{X} = \begin{pmatrix} 1-1.75 & -1-0.5 & 3-1.75 \\ 1-1.75 & 0-0.5 & 3-1.75 \\ 2-1.75 & 3-0.5 & 0-1.75 \\ 3-1.75 & 0-0.5 & 1-1.75 \end{pmatrix} = \begin{pmatrix} -0.75 & -1.5 & 1.25 \\ -0.75 & -0.5 & 1.25 \\ 0.25 & 2.5 & -1.75 \\ 1.25 & -0.5 & -0.75 \end{pmatrix}.$$

La matriz de datos centrados admite un cálculo matricial sencillo. Sea H_n la matriz que se obtiene restando $1/n$ a todas las entradas de la matriz identidad I_n:

$$H_n = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} - \frac{1}{n}\begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{pmatrix} = \begin{pmatrix} 1-\frac{1}{n} & -\frac{1}{n} & \dots & -\frac{1}{n} \\ -\frac{1}{n} & 1-\frac{1}{n} & \dots & -\frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{n} & -\frac{1}{n} & \dots & 1-\frac{1}{n} \end{pmatrix}.$$

A esta matriz H_n se la llama la *matriz centralizadora* de orden n. Se tiene entonces el resultado siguiente:

Teorema 4.1. $\widetilde{X} = H_n \cdot X$.

Demostración: Sea 1_n el vector fila formado por n unos,

$$1_n = (\overbrace{1, 1, \ldots, 1}^{n}).$$

Tenemos entonces que

$$1_n^t \cdot \overline{x} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \cdot (\overline{x}_{\bullet 1}, \overline{x}_{\bullet 2}, \ldots, \overline{x}_{\bullet p}) = \begin{pmatrix} \overline{x}_{\bullet 1} & \overline{x}_{\bullet 2} & \ldots & \overline{x}_{\bullet p} \\ \overline{x}_{\bullet 1} & \overline{x}_{\bullet 2} & \ldots & \overline{x}_{\bullet p} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{x}_{\bullet 1} & \overline{x}_{\bullet 2} & \ldots & \overline{x}_{\bullet p} \end{pmatrix}$$

y por lo tanto

$$\widetilde{X} = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix} - \begin{pmatrix} \overline{x}_{\bullet 1} & \overline{x}_{\bullet 2} & \ldots & \overline{x}_{\bullet p} \\ \overline{x}_{\bullet 1} & \overline{x}_{\bullet 2} & \ldots & \overline{x}_{\bullet p} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{x}_{\bullet 1} & \overline{x}_{\bullet 2} & \ldots & \overline{x}_{\bullet p} \end{pmatrix} = X - 1_n^t \cdot \overline{x}.$$

Por otro lado, cuando multiplicamos 1_n por una matriz de n filas, obtenemos un vector fila formado por las sumas de sus columnas:

$$(1, 1, \ldots, 1) \cdot \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix} = \left(x_{11} + x_{21} + \cdots + x_{n1}, \ldots, x_{1p} + x_{2p} + \cdots + x_{np}\right).$$

Por consiguiente,

$$\begin{aligned} \frac{1}{n} \cdot 1_n \cdot X &= \frac{1}{n} \cdot (1, \ldots, 1) \cdot \begin{pmatrix} x_{11} & \ldots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \ldots & x_{np} \end{pmatrix} \\ &= \frac{1}{n}\left(x_{11} + \cdots + x_{n1}, \ldots, x_{1p} + \cdots + x_{np}\right) = (\overline{x}_{\bullet 1}, \ldots, \overline{x}_{\bullet p}) = \overline{x}. \end{aligned}$$

Combinando las igualdades

$$\widetilde{X} = X - 1_n^t \cdot \overline{x} \quad \text{y} \quad \overline{x} = \frac{1}{n} \cdot 1_n \cdot X$$

concluimos que

$$\widetilde{X} = X - \frac{1}{n} \cdot 1_n^t \cdot 1_n \cdot X = (I_n - \frac{1}{n} \cdot 1_n^t \cdot 1_n)X.$$

Finalmente,

$$I_n - \frac{1}{n} \cdot 1_n^t \cdot 1_n = \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{pmatrix} - \frac{1}{n} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \cdot (1, 1, \ldots, 1) = \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{pmatrix} - \frac{1}{n} \begin{pmatrix} 1 & 1 & \ldots & 1 \\ 1 & 1 & \ldots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \ldots & 1 \end{pmatrix} = H_n$$

y obtenemos la identidad del enunciado. □

Las matrices centralizadoras satisfacen las propiedades siguientes, que se pueden comprobar fácilmente:

- H_n es una matriz *idempotente*, es decir, $H_n \cdot H_n = H_n$.
- H_n es simétrica, tiene rango $n-1$ y $H_n \cdot 1_n^t = 0$.

Ejemplo 4.3. Para centrar matricialmente la tabla de datos del ejemplo anterior, usamos la matriz centralizadora

$$H_4 = \begin{pmatrix} 3/4 & -1/4 & -1/4 & -1/4 \\ -1/4 & 3/4 & -1/4 & -1/4 \\ -1/4 & -1/4 & 3/4 & -1/4 \\ -1/4 & -1/4 & -1/4 & 3/4 \end{pmatrix} = \begin{pmatrix} 0.75 & -0.25 & -0.25 & -0.25 \\ -0.25 & 0.75 & -0.25 & -0.25 \\ -0.25 & -0.25 & 0.75 & -0.25 \\ -0.25 & -0.25 & -0.25 & 0.75 \end{pmatrix}$$

y obtenemos

$$\widetilde{X} = H_4 \cdot X = \begin{pmatrix} 0.75 & -0.25 & -0.25 & -0.25 \\ -0.25 & 0.75 & -0.25 & -0.25 \\ -0.25 & -0.25 & 0.75 & -0.25 \\ -0.25 & -0.25 & -0.25 & 0.75 \end{pmatrix} \cdot \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix} = \begin{pmatrix} -0.75 & -1.5 & 1.25 \\ -0.75 & -0.5 & 1.25 \\ 0.25 & 2.5 & -1.75 \\ 1.25 & -0.5 & -0.75 \end{pmatrix}.$$

Dado un vector de datos formado por una muestra de una variable cuantitativa, su *vector de datos tipificados* es el vector que se obtiene restando a cada entrada la media aritmética del vector y dividiendo el resultado por su desviación típica. De esta manera, se obtiene un vector de datos de media aritmética 0 y varianza 1. Tipificar un vector de datos es conveniente cuando se quiere trabajar con estos datos sin que influyan ni su media ni las unidades en los que están medidos: al dividir por su desviación típica, los valores resultantes son adimensionales. Por lo tanto, tipificar las variables de una tabla de datos permite compararlas dejando de lado las diferencias que pueda haber entre sus valores medios o sus varianzas.

La *matriz tipificada* de una matriz de datos X es la matriz Z que se obtiene tipificando cada columna; es decir, para tipificar una matriz de datos X, restamos a cada columna su media y a continuación dividimos cada columna por la desviación típica de la columna original en X (que coincide con la desviación típica de la columna «centrada»):

$$Z = \begin{pmatrix} \frac{x_{11}-\overline{x}_{\bullet 1}}{s_1} & \frac{x_{12}-\overline{x}_{\bullet 2}}{s_2} & \cdots & \frac{x_{1p}-\overline{x}_{\bullet p}}{s_p} \\ \frac{x_{21}-\overline{x}_{\bullet 1}}{s_1} & \frac{x_{22}-\overline{x}_{\bullet 2}}{s_2} & \cdots & \frac{x_{2p}-\overline{x}_{\bullet p}}{s_p} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{x_{n1}-\overline{x}_{\bullet 1}}{s_1} & \frac{x_{n2}-\overline{x}_{\bullet 2}}{s_2} & \cdots & \frac{x_{np}-\overline{x}_{\bullet p}}{s_p} \end{pmatrix}.$$

Sea D la matriz diagonal $p \times p$ que tiene en su diagonal principal las desviaciones típicas de las columnas correspondientes de X; su inversa D^{-1} es la matriz diagonal $p \times p$ que

tiene en su diagonal principal los inversos de estas desviaciones típicas:

$$D = \begin{pmatrix} s_1 & 0 & \dots & 0 \\ 0 & s_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & s_p \end{pmatrix},\ D^{-1} = \begin{pmatrix} \frac{1}{s_1} & 0 & \dots & 0 \\ 0 & \frac{1}{s_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{s_p} \end{pmatrix}.$$

Se tiene entonces el resultado siguiente, que nos da una expresión matricial sencilla de la matriz tipificada.

Teorema 4.2. $Z = H_n \cdot X \cdot D^{-1}$.

Demostración: Z se obtiene centrando X y luego dividiendo cada columna por s_i, lo que corresponde a $Z = \widetilde{X} \cdot D^{-1} = H_n \cdot X \cdot D^{-1}$. □

Ejemplo 4.4. Vamos a tipificar a mano la tabla de datos

$$X = \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}$$

del Ejemplo 4.1. Ya hemos calculado las medias de sus columnas con `R` en el Ejemplo 4.1 y a mano en el Ejemplo 4.2. Por lo que refiere a las varianzas de sus columnas, también las hemos calculado con `R` en el Ejemplo 4.1, y son

$$s_1^2 = \frac{1}{4}(1^2+1^2+2^2+3^2) - \left(\frac{7}{4}\right)^2 = \frac{11}{16} = 0.6875,\ s_2^2 = \frac{1}{4}((-1)^2+3^2) - \left(\frac{1}{2}\right)^2 = \frac{9}{4} = 2.25$$

$$s_3^2 = \frac{1}{4}(3^2+3^2+1^2) - \left(\frac{7}{4}\right)^2 = \frac{27}{16} = 1.6875$$

Así pues,

$$D^{-1} = \begin{pmatrix} \frac{1}{\sqrt{11/16}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{9/4}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{27/16}} \end{pmatrix} = \begin{pmatrix} \frac{4}{\sqrt{11}} & 0 & 0 \\ 0 & \frac{2}{3} & 0 \\ 0 & 0 & \frac{4}{3\sqrt{3}} \end{pmatrix}$$

y por lo tanto

$$\begin{aligned} Z &= H_4 \cdot X \cdot D^{-1} \\ &= \begin{pmatrix} 3/4 & -1/4 & -1/4 & -1/4 \\ -1/4 & 3/4 & -1/4 & -1/4 \\ -1/4 & -1/4 & 3/4 & -1/4 \\ -1/4 & -1/4 & -1/4 & 3/4 \end{pmatrix} \cdot \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 4/\sqrt{11} & 0 & 0 \\ 0 & 2/3 & 0 \\ 0 & 0 & 4/(3\sqrt{3}) \end{pmatrix} \\ &= \begin{pmatrix} -3/\sqrt{11} & -1 & 5/(3\sqrt{3}) \\ -3/\sqrt{11} & -1/3 & 5/(3\sqrt{3}) \\ 1/\sqrt{11} & 5/3 & -7/(3\sqrt{3}) \\ 5/\sqrt{11} & -1/3 & -3/(3\sqrt{3}) \end{pmatrix} \end{aligned}$$

La manera más sencilla de aplicar con `R` una transformación lineal a una tabla de datos X, y en particular de centrarla o tipificarla, es usando la instrucción

```
scale(X, center=..., scale=...)
```

donde:

- `X` puede ser tanto una matriz como un *data frame*; el resultado será siempre una matriz.
- El valor del parámetro `center` es el vector que restamos a sus columnas, en el sentido de que cada entrada de este vector se restará a todas las entradas de la columna correspondiente. Su valor por defecto (que no es necesario especificar, aunque también se puede especificar con `center=TRUE`) es el vector $\overline{X}$ de medias de X; para especificar que no se reste nada, podemos usar `center=FALSE`.
- El valor del parámetro `scale` es el vector por el que dividimos las columnas de X: cada columna se divide por la entrada correspondiente de este vector. Su valor por defecto (de nuevo, se puede especificar igualando el parámetro a `TRUE`) es el vector $\widetilde{s}_X$ de desviaciones típicas *muestrales*; para especificar que no se divida por nada, podemos usar `scale=FALSE`.

En particular, la instrucción `scale(X)` centra la tabla de datos X y divide sus columnas por sus *desviaciones típicas muestrales*; por lo tanto, no la tipifica según nuestra definición, ya que no las divide por sus desviaciones típicas «verdaderas».

Ejemplo 4.5. Vamos a centrar la tabla de datos X del Ejemplo 4.1.

```
> X
  V1 V2 V3
1  1 -1  3
2  1  0  3
3  2  3  0
4  3  0  1
> X_centrada=scale(X, center=TRUE, scale=FALSE)
> X_centrada
        V1   V2    V3
[1, ] -0.75 -1.5  1.25
[2, ] -0.75 -0.5  1.25
[3, ]  0.25  2.5 -1.75
[4, ]  1.25 -0.5 -0.75
attr(, "scaled:center")
  V1   V2   V3
1.75 0.50 1.75
```

Observad la estructura del resultado: en primer lugar nos da la matriz centrada, y a continuación nos dice que tiene un atributo llamado `"scaled:center"` cuyo valor es el vector

usado para centrarla. Este atributo no interferirá para nada en las operaciones que se realicen con la matriz centrada, pero, si os molesta, recordad que se puede eliminar sustituyendo el resultado de centrar la matriz en los puntos suspensivos de la instrucción siguiente:

`attr(..., "scaled:center")=NULL`.

```
> attr(X_centrada, "scaled:center")=NULL
> X_centrada
        V1   V2    V3
[1, ] -0.75 -1.5  1.25
[2, ] -0.75 -0.5  1.25
[3, ]  0.25  2.5 -1.75
[4, ]  1.25 -0.5 -0.75
```

Como ya hemos avisado, para tipificar esta tabla de datos *no* podemos hacer lo siguiente:

```
> X_tip=scale(X)
> X_tip
             V1         V2         V3
[1, ] -0.7833495 -0.8660254  0.8333333
[2, ] -0.7833495 -0.2886751  0.8333333
[3, ]  0.2611165  1.4433757 -1.1666667
[4, ]  1.3055824 -0.2886751 -0.5000000
attr(, "scaled:center")
  V1   V2   V3
1.75 0.50 1.75
attr(, "scaled:scale")
       V1        V2        V3
0.9574271 1.7320508 1.5000000
```

Para hacerlo bien en base a la definición que hemos dado, tenemos dos opciones. Una posibilidad es multiplicar la matriz anterior por $\sqrt{n/(n-1)}$, donde n es el número de filas de la tabla.[1]

```
> n=dim(X)[1]  #Número de filas de X
> n
[1] 4
> X_tip=scale(X)*sqrt(n/(n-1))
> X_tip
             V1         V2         V3
[1, ] -0.9045340 -1.0000000  0.9622504
[2, ] -0.9045340 -0.3333333  0.9622504
[3, ]  0.3015113  1.6666667 -1.3471506
[4, ]  1.5075567 -0.3333333 -0.5773503
```

[1] Como $\widetilde{s}_X = \sqrt{\frac{n}{n-1}} \cdot s_X$, se tiene que $\frac{1}{s_X} = \sqrt{\frac{n}{n-1}} \cdot \frac{1}{\widetilde{s}_X}$; por lo tanto, si queríamos dividir por s_X y `scale(X)` ha dividido por $\widetilde{s}_X$, basta multiplicar su resultado por $\sqrt{\frac{n}{n-1}}$.

```
attr(, "scaled:center")
  V1   V2   V3
1.75 0.50 1.75
attr(, "scaled:scale")
       V1        V2        V3
0.9574271 1.7320508 1.5000000
```

Otra posibilidad es usar, como valor del parámetro `scale`, el vector s_X de desviaciones típicas de las columnas.

```
> sd_ver=function(x){sqrt(var(x)*(length(x)-1)/length(x))}
> X_dtv=sapply(X, sd_ver) #Desviaciones típicas "verdaderas"
> X_tip1=scale(X, scale=X_dtv) #Escalamos dividiendo las
    columnas por X_dtv
> X_tip1
             V1         V2         V3
[1, ] -0.9045340 -1.0000000  0.9622504
[2, ] -0.9045340 -0.3333333  0.9622504
[3, ]  0.3015113  1.6666667 -1.3471506
[4, ]  1.5075567 -0.3333333 -0.5773503
attr(, "scaled:center")
  V1   V2   V3
1.75 0.50 1.75
attr(, "scaled:scale")
       V1        V2        V3
0.8291562 1.5000000 1.2990381
```

Observaréis que la matriz resultante es la misma, pero el atributo que indica el vector por el que hemos dividido las columnas es diferente: en este caso, es el de desviaciones típicas. Ahora, en ambos casos, podemos usar la función `attr` para eliminar los dos atributos, `"scaled:center"` y `"scaled:scale"`, que se han añadido a la matriz tipificada.

```
> attr(X_tip, "scaled:center")=NULL
> attr(X_tip, "scaled:scale")=NULL
> X_tip
             V1         V2         V3
[1, ] -0.9045340 -1.0000000  0.9622504
[2, ] -0.9045340 -0.3333333  0.9622504
[3, ]  0.3015113  1.6666667 -1.3471506
[4, ]  1.5075567 -0.3333333 -0.5773503
```

4.4. Covarianzas y correlaciones

La *covarianza* entre dos variables es una medida de la propensión que tienen ambas variables a variar conjuntamente. Cuando la covarianza es positiva, si una de las dos variables crece o decrece, la otra tiene el mismo comportamiento; en cambio, cuando la covarianza

es negativa, esta tendencia se invierte: si una variable crece, la otra decrece y viceversa. Por desgracia, interpretar el valor de la covarianza más allá de su signo es difícil, por lo que introduciremos una versión «normalizada» de la misma, la *correlación de Pearson*, que mide de manera más precisa la relación lineal entre dos variables.

La covarianza generaliza la varianza, en el sentido de que la varianza de una variable es su covarianza consigo misma. Y como en el caso de la varianza, definiremos dos versiones de la covarianza: la «*verdadera*» y la *muestral*. La diferencia estará de nuevo en el denominador.

Formalmente, la *covarianza* de las variables $x_{\bullet i}$ y $x_{\bullet j}$ de una matriz de datos X es

$$s_{ij} = \frac{1}{n}\sum_{k=1}^{n}\big((x_{ki}-\overline{x}_{\bullet i})(x_{kj}-\overline{x}_{\bullet j})\big) = \frac{1}{n}\Big(\sum_{k=1}^{n} x_{ki}x_{kj}\Big) - \overline{x}_{\bullet i}\overline{x}_{\bullet j},$$

y su *covarianza muestral* es

$$\widetilde{s}_{ij} = \frac{1}{n-1}\sum_{k=1}^{n}\big((x_{ki}-\overline{x}_{\bullet i})(x_{kj}-\overline{x}_{\bullet j})\big) = \frac{n}{n-1}s_{ij}.$$

El estadístico s_{ij} es el estimador máximo verosímil de la covarianza σ_{ij} de las variables aleatorias X_i y X_j cuando su distribución conjunta es normal bivariante,[2] mientras que $\tilde{s}_{ij}$ es siempre un estimador insesgado de dicha covarianza.

Es inmediato comprobar a partir de sus definiciones que ambas covarianzas son simétricas, y que la covarianza de una variable consigo misma es su varianza:

$$s_{ij} = s_{ji}, \quad \widetilde{s}_{ij} = \widetilde{s}_{ji}, \quad s_{ii} = s_i^2, \quad \widetilde{s}_{ii} = \widetilde{s}_i^2.$$

Ejemplo 4.6. La covarianza de las dos primeras columnas de la matriz de datos

$$X = \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}$$

del Ejemplo 4.1 se calcularía de la manera siguiente:

$$s_{12} = \frac{1}{4}(1\cdot(-1)+1\cdot 0+2\cdot 3+3\cdot 0) - 1.75\cdot 0.5 = 1.25 - 0.875 = 0.375$$

Su covarianza muestral se obtendría multiplicando por $4/3$ este valor:

$$\widetilde{s}_{12} = \frac{4}{3}s_{12} = 0.5.$$

La covarianza *muestral* de dos vectores numéricos de la misma longitud n se puede calcular con `R` mediante la función `cov`. Para obtener su covarianza «verdadera», hay que multiplicar el resultado de `cov` por $(n-1)/n$.

[2] `http://es.wikipedia.org/wiki/Distribución_normal_multivariante`

```
> X
  V1 V2 V3
1  1 -1  3
2  1  0  3
3  2  3  0
4  3  0  1
> cov(X$V1, X$V2)   #Covarianza MUESTRAL
[1] 0.5
> (3/4)*cov(X$V1, X$V2)  #Covarianza "verdadera"
[1] 0.375
```

Queremos recalcar que, como en el caso de la varianza con `var`, `R` calcula con `cov` la versión muestral de la covarianza.

Las *matrices de covarianzas* y de *covarianzas muestrales* de una tabla de datos X son, respectivamente,

$$S = \begin{pmatrix} s_{11} & s_{12} & \dots & s_{1p} \\ s_{21} & s_{22} & \dots & s_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & \dots & s_{pp} \end{pmatrix}, \ \widetilde{S} = \begin{pmatrix} \widetilde{s}_{11} & \widetilde{s}_{12} & \dots & \widetilde{s}_{1p} \\ \widetilde{s}_{21} & \widetilde{s}_{22} & \dots & \widetilde{s}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{s}_{p1} & \widetilde{s}_{p2} & \dots & \widetilde{s}_{pp} \end{pmatrix},$$

donde cada s_{ij} y cada $\widetilde{s}_{ij}$ son, respectivamente, la covarianza y la covarianza muestral de las correspondientes columnas $x_{\bullet i}$ y $x_{\bullet j}$. Estas matrices de covarianzas miden la tendencia a la variabilidad conjunta de los datos de X. Si el vector de variables $\underline{X}$ tiene distribución normal multivariante, la matriz S es un estimador máximo verosímil de la matriz de covarianzas Σ y $\widetilde{S}$ es un estimador insesgado de Σ.

Ambas son simétricas y tienen todos sus valores propios $\geqslant 0$. Además, se tiene el resultado siguiente, que permite su cálculo matricial.

Teorema 4.3. $S = \dfrac{1}{n}\widetilde{X}^t \cdot \widetilde{X} = \dfrac{1}{n} X^t \cdot H_n \cdot X.$

Ejemplo 4.7. Continuemos con la matriz

$$X = \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}$$

del Ejemplo 4.1. Vamos a calcular a mano su matriz de covarianzas y comprobar que coincide con la dada por la fórmula anterior. Para realizar los cálculos a mano, es útil organizar los datos y los cálculos intermedios necesarios en una tabla como la siguiente:

i	$x_{\bullet 1}$	$x_{\bullet 2}$	$x_{\bullet 3}$	$x_{\bullet 1}^2$	$x_{\bullet 2}^2$	$x_{\bullet 3}^2$	$x_{\bullet 1}x_{\bullet 2}$	$x_{\bullet 1}x_{\bullet 3}$	$x_{\bullet 2}x_{\bullet 3}$
1	1	-1	3	1	1	9	-1	3	-3
2	1	0	3	1	0	9	0	3	0
3	2	3	0	4	9	0	6	0	0
4	3	0	1	9	0	1	0	3	0
Suma	7	2	7	15	10	19	5	9	-3
Media	7/4	2/4	7/4	15/4	10/4	19/4	5/4	9/4	$-3/4$

Así tenemos que

$$s_1^2 = \frac{1}{4}\Big(\sum_{i=1}^{4} x_{i1}^2\Big) - \overline{x}_{\bullet 1}^2 = \frac{15}{4} - \left(\frac{7}{4}\right)^2 = \frac{11}{16} = 0.6875$$
$$s_2^2 = \frac{1}{4}\Big(\sum_{i=1}^{4} x_{i2}^2\Big) - \overline{x}_{\bullet 2}^2 = \frac{10}{4} - \left(\frac{2}{4}\right)^2 = \frac{9}{4} = 2.25$$
$$s_3^2 = \frac{1}{4}\Big(\sum_{i=1}^{4} x_{i3}^2\Big) - \overline{x}_{\bullet 3}^2 = \frac{19}{4} - \left(\frac{7}{4}\right)^2 = \frac{27}{16} = 1.6875$$
$$s_{12} = \frac{1}{4}\Big(\sum_{i=1}^{n} x_{i1}x_{i2}\Big) - \overline{x}_{\bullet 1}\overline{x}_{\bullet 2} = \frac{5}{4} - \frac{7}{4}\cdot\frac{2}{4} = \frac{3}{8} = 0.375$$
$$s_{13} = \frac{1}{4}\Big(\sum_{i=1}^{n} x_{i1}x_{i3}\Big) - \overline{x}_{\bullet 1}\overline{x}_{\bullet 3} = \frac{9}{4} - \frac{7}{4}\cdot\frac{7}{4} = -\frac{13}{16} = -0.8125$$
$$s_{23} = \frac{1}{4}\Big(\sum_{i=1}^{n} x_{i2}x_{i3}\Big) - \overline{x}_{\bullet 2}\overline{x}_{\bullet 3} = \frac{-3}{4} - \frac{2}{4}\cdot\frac{7}{4} = -\frac{13}{8} = -1.625$$

Y por lo tanto, la matriz de covarianzas es

$$S = \begin{pmatrix} 0.6875 & 0.375 & -0.8125 \\ 0.375 & 2.25 & -1.625 \\ -0.8125 & -1.625 & 1.6875 \end{pmatrix}$$

También se puede obtener aplicando el Teorema 4.3:

$$\begin{aligned} S &= \frac{1}{4}X^t \cdot H_4 \cdot X \\ &= \frac{1}{4}\begin{pmatrix} 1 & 1 & 2 & 3 \\ -1 & 0 & 3 & 0 \\ 3 & 3 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 3/4 & -1/4 & -1/4 & -1/4 \\ -1/4 & 3/4 & -1/4 & -1/4 \\ -1/4 & -1/4 & 3/4 & -1/4 \\ -1/4 & -1/4 & -1/4 & 3/4 \end{pmatrix} \cdot \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix} = \dots \end{aligned}$$

```
> H4=diag(4)-(1/4)*matrix(1,nrow=4,ncol=4) #Matriz
    centralizadora
> H4
      [,1]   [,2]   [,3]   [,4]
[1,]  0.75 -0.25 -0.25 -0.25
```

```
[2,] -0.25  0.75 -0.25 -0.25
[3,] -0.25 -0.25  0.75 -0.25
[4,] -0.25 -0.25 -0.25  0.75
> Xm=as.matrix(X)  #El dataframe X, convertido en matriz
> S=(1/4)*t(Xm)%*%H4%*%Xm
> S
         V1      V2      V3
V1  0.6875  0.375 -0.8125
V2  0.3750  2.250 -1.6250
V3 -0.8125 -1.625  1.6875
```

La matriz de covarianzas muestrales de X se calcula aplicando la función `cov` al *data frame* o a la matriz que contenga dicha tabla. Para obtener su matriz de covarianzas «verdaderas», es suficiente multiplicar el resultado de `cov` por $(n-1)/n$, donde n es el número de filas de X.

```
> n=dim(X)[1]
> cov(X)  #Matriz de covarianzas muestrales
            V1        V2        V3
V1  0.9166667  0.500000 -1.083333
V2  0.5000000  3.000000 -2.166667
V3 -1.0833333 -2.166667  2.250000
> ((n-1)/n)*cov(X)  #Matriz de covarianzas "verdaderas"
         V1      V2      V3
V1  0.6875  0.375 -0.8125
V2  0.3750  2.250 -1.6250
V3 -0.8125 -1.625  1.6875
```

Como la matriz de covarianzas como medida de variabilidad es difícil de interpretar, debido a que no es una única cantidad sino toda una matriz, interesa cuantificar esta variabilidad mediante un único índice. No hay consenso sobre este índice, y entre los que se han propuesto destacamos:

- La *varianza total* de X: la suma de las varianzas de sus columnas.
- La *varianza media* de X: la media de las varianzas de sus columnas, es decir, la varianza total partida por el número de columnas.
- La *varianza generalizada* de X: el determinante de su matriz de covarianzas.
- La *desviación típica generalizada* de X: la raíz cuadrada positiva de su varianza generalizada.

La *correlación lineal de Pearson* (o, de ahora en adelante, simplemente *correlación*) de las variables $x_{\bullet i}$ y $x_{\bullet j}$ de X es

$$r_{ij} = \frac{s_{ij}}{s_i s_j}.$$

Observad que

$$\frac{\widetilde{s}_{ij}}{\widetilde{s}_i \cdot \widetilde{s}_j} = \frac{\frac{n}{n-1} \cdot s_{ij}}{\sqrt{\frac{n}{n-1}} \cdot s_i \cdot \sqrt{\frac{n}{n-1}} \cdot s_j} = \frac{s_{ij}}{s_i \cdot s_j} = r_{ij},$$

y, por lo tanto, esta correlación se puede calcular también a partir de las versiones muestrales de la covarianza y las desviaciones típicas por medio de la misma fórmula.

La correlación r_{ij} estima el parámetro poblacional $\rho_{ij} = Cor(X_i, X_j)$, y tiene las propiedades siguientes:

- Es simétrica: $r_{ij} = r_{ji}$.
- $-1 \leqslant r_{ij} \leqslant 1$.
- $r_{ii} = 1$.
- r_{ij} tiene el mismo signo que s_{ij}.
- $r_{ij} = \pm 1$ si y, sólo si, existe una relación lineal perfecta entre las variables $x_{\bullet i}$ y $x_{\bullet j}$: es decir, si, y sólo si, existen valores $a, b \in \mathbb{R}$ tales que

  $$\begin{pmatrix} x_{1j} \\ \vdots \\ x_{nj} \end{pmatrix} = a \cdot \begin{pmatrix} x_{1i} \\ \vdots \\ x_{ni} \end{pmatrix} + b.$$

  La pendiente a de esta relación lineal tiene el mismo signo que r_{ij}.
- El coeficiente de determinación R^2 de la regresión lineal por mínimos cuadrados de $x_{\bullet j}$ respecto de $x_{\bullet i}$ es igual al cuadrado de su correlación, r_{ij}^2; por lo tanto, cuánto más se aproxime el valor absoluto de r_{ij} a 1, más se acercan las variables $x_{\bullet i}$ y $x_{\bullet j}$ a depender linealmente la una de la otra.

Así pues, la correlación entre dos variables viene a ser una covarianza «normalizada», ya que, como vemos, su valor está entre -1 y 1, y mide la tendencia de las variables a estar relacionadas según una función lineal. En concreto, cuanto más se acerca la correlación a 1 (respectivamente, a -1), más se acerca una (cualquiera) de las variables a ser función lineal creciente (respectivamente, decreciente) de la otra.

Con `R`, la correlación de Pearson de dos vectores se puede calcular por medio de la función `cor`.

Ejemplo 4.8. En ejemplos anteriores hemos calculado la covarianza y las varianzas de las dos primeras columnas de la matriz de datos

$$X = \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}.$$

Hemos obtenido los valores siguientes

$$s_{12} = 0.375, \quad s_1 = \frac{\sqrt{11}}{4} = 0.82916, \quad s_2 = \frac{3}{2} = 1.5.$$

Por lo tanto, su correlación es

$$r_{12} = \frac{0.375}{0.82916 \cdot 1.5} = 0.3015.$$

Ahora vamos a calcularla con `R`, y aprovecharemos para confirmar su relación con el valor de R^2 de la regresión lineal de la segunda columna respecto de la primera.

```
> X
  V1 V2 V3
1  1 -1  3
2  1  0  3
3  2  3  0
4  3  0  1
> cor(X$V1, X$V2)
[1] 0.3015113
> cor(X$V1, X$V2)^2
[1] 0.09090909
> summary(lm(X$V2~X$V1))$r.squared
[1] 0.09090909
```

La *matriz de correlaciones* (*de Pearson*) de X es

$$R = \begin{pmatrix} 1 & r_{12} & \dots & r_{1p} \\ r_{21} & 1 & \dots & r_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ r_{p1} & r_{p2} & \dots & 1 \end{pmatrix}$$

donde cada r_{ij} es la correlación de las columnas correspondientes de X.

Esta matriz de correlaciones tiene siempre determinante $|R| \leqslant 1$ y todos sus valores propios $\geqslant 0$, y se puede calcular de forma matricial de la manera siguiente. Recordemos que

$$D^{-1} = \begin{pmatrix} \frac{1}{s_1} & 0 & \dots & 0 \\ 0 & \frac{1}{s_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{s_p} \end{pmatrix}.$$

Teorema 4.4. $R = D^{-1} \cdot S \cdot D^{-1}$.

Ejemplo 4.9. Continuemos con el Ejemplo 4.1, donde, recordemos,

$$X = \begin{pmatrix} 1 & -1 & 3 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}.$$

En otros ejemplos ya hemos calculado

$$D^{-1} = \begin{pmatrix} 4/\sqrt{11} & 0 & 0 \\ 0 & 2/3 & 0 \\ 0 & 0 & 4/(3\sqrt{3}) \end{pmatrix}, \; S = \begin{pmatrix} 11/16 & 3/8 & -13/16 \\ 3/8 & 9/4 & -13/8 \\ -13/16 & -13/8 & 27/16 \end{pmatrix},$$

y por lo tanto su matriz de correlaciones es

$$\begin{aligned} R &= \begin{pmatrix} \frac{4}{\sqrt{11}} & 0 & 0 \\ 0 & \frac{2}{3} & 0 \\ 0 & 0 & \frac{4}{3\sqrt{3}} \end{pmatrix} \cdot \begin{pmatrix} \frac{11}{16} & \frac{3}{8} & -\frac{13}{16} \\ \frac{3}{8} & \frac{9}{4} & -\frac{13}{8} \\ -\frac{13}{16} & -\frac{13}{8} & \frac{27}{16} \end{pmatrix} \cdot \begin{pmatrix} \frac{4}{\sqrt{11}} & 0 & 0 \\ 0 & \frac{2}{3} & 0 \\ 0 & 0 & \frac{4}{3\sqrt{3}} \end{pmatrix} \\ &= \begin{pmatrix} 1 & \frac{1}{\sqrt{11}} & -\frac{13}{3\sqrt{33}} \\ \frac{1}{\sqrt{11}} & 1 & -\frac{13}{9\sqrt{3}} \\ -\frac{13}{3\sqrt{33}} & -\frac{13}{9\sqrt{3}} & 1 \end{pmatrix} \end{aligned}$$

La matriz de correlaciones de una tabla de datos se puede calcular con `R` con la misma instrucción `cor`.

```
> cor(X)
           V1          V2          V3
V1  1.0000000  0.3015113 -0.7543365
V2  0.3015113  1.0000000 -0.8339504
V3 -0.7543365 -0.8339504  1.0000000
```

Comprobemos que da lo mismo que el producto de matrices del Teorema 4.4.

```
> sd_ver=function(x){sqrt(var(x)*(length(x)-1)/length(x))}
> X.dt=sapply(X,sd_ver)
> Dm=diag(1/X.dt)   #Matriz diagonal D de inversas de
   desviaciones típicas
> S=(3/4)*cov(X)
> Dm%*%S%*%Dm
           [,1]        [,2]        [,3]
[1,]  1.0000000  0.3015113 -0.7543365
[2,]  0.3015113  1.0000000 -0.8339504
[3,] -0.7543365 -0.8339504  1.0000000
```

R también dispone de la función `cov2cor` que aplicada a la matriz de covarianzas (muestrales o no) da la matriz de correlaciones:

```
> cov2cor(S)
           [,1]       [,2]       [,3]
[1,]  1.0000000  0.3015113 -0.7543365
[2,]  0.3015113  1.0000000 -0.8339504
[3,] -0.7543365 -0.8339504  1.0000000
```

Se tiene el teorema siguiente, que se puede demostrar mediante un simple, aunque farragoso, cálculo algebraico:

Teorema 4.5. La matriz de correlaciones de X es igual a la matriz de covarianzas de su matriz tipificada.

La importancia de este resultado es que, si la tabla de datos es muy grande, suele ser más eficiente calcular la matriz de covarianzas de su matriz tipificada que la matriz de correlaciones de la tabla original. Comprobemos que el teorema es cierto para nuestra matriz de datos:

```
> cov(X)
           V1         V2         V3
V1  0.9166667  0.500000 -1.083333
V2  0.5000000  3.000000 -2.166667
V3 -1.0833333 -2.166667  2.250000
> n=dim(X)[1]
> X_tip=scale(X)*sqrt(n/(n-1))
> cov(X_tip)*(n-1)/n
           V1         V2         V3
V1  1.0000000  0.3015113 -0.7543365
V2  0.3015113  1.0000000 -0.8339504
V3 -0.7543365 -0.8339504  1.0000000
```

Cuando se calcula la covarianza o la correlación de dos vectores que contienen valores `NA`, lo usual es no tenerlos en cuenta: es decir, si un vector contiene un `NA` en una posición, se eliminan de los dos vectores sus entradas en dicha posición. De esta manera, se tomaría como covarianza de

$$\begin{pmatrix} 1 \\ 2 \\ NA \\ 4 \\ 6 \\ 2 \end{pmatrix} \text{ y } \begin{pmatrix} 2 \\ 4 \\ -3 \\ 5 \\ 7 \\ NA \end{pmatrix}$$

la de

$$\begin{pmatrix} 1 \\ 2 \\ 4 \\ 6 \end{pmatrix} \text{ y } \begin{pmatrix} 2 \\ 4 \\ 5 \\ 7 \end{pmatrix}.$$

Al aplicar `cov` o `cor` a un par de vectores que contengan `NA`, se obtiene, por defecto, `NA`. Si se quiere que `R` calcule el valor sin tener en cuenta los `NA`, se ha de especificar añadiendo el parámetro `use="complete.obs"` (que le indica que ha de usar las observaciones completas, es decir, las posiciones que no tienen `NA` en ninguno de los dos vectores).

```
> x=c(1,2,NA,4,6,2)
> y=c(2,4,-3,5,7,NA)
> x1=c(1,2,4,6)  #Quitamos las entradas 3a y 6a
> x2=c(2,4,5,7)  #Quitamos las entradas 3a y 6a
> cov(x, y)
[1] NA
> cov(x, y, use="complete.obs")
[1] 4.5
> cov(x1, x2)
[1] 4.5
> cor(x, y)
[1] NA
> cor(x, y, use="complete.obs")
[1] 0.9749135
> cor(x1, x2)
[1] 0.9749135
```

Al calcular las matrices de covarianzas, covarianzas muestrales o correlaciones de una tabla de datos que contenga `NA`, se suele seguir una de las dos estrategias siguientes, según lo que interese al usuario:

- Para cada par de columnas, se calcula su covarianza o su correlación con la estrategia explicada más arriba para dos vectores, obviando el hecho de que forman parte de una tabla de datos mayor; es decir, al efectuar el cálculo para cada par de columnas concreto, se eliminan de cada una de ellas sus entradas `NA` y aquellas en cuya fila la otra tiene un `NA`. Esta opción se especifica dentro de la función `cov` o `cor` con el parámetro `use="pairwise.complete.obs"`.
- Antes de nada, se eliminan las filas de la tabla que contienen algún `NA` en alguna columna, dejando solo en la tabla las filas «completas», las que no contienen ningún `NA`. Luego se calcula la matriz de covarianzas o de correlaciones de la tabla resultante. Esta opción se especifica con el parámetro `use="complete.obs"`.

Veamos un ejemplo:

```
> X=cbind(c(1,2,NA,4,6,2), c(2,4,-3,5,7,NA), c(-2,1,0,2,3,0))
```

```
> X
     [, 1] [, 2] [, 3]
[1, ]     1     2   -2
[2, ]     2     4    1
[3, ]    NA    -3    0
[4, ]     4     5    2
[5, ]     6     7    3
[6, ]     2    NA    0
> cov(X)
     [, 1] [, 2]     [, 3]
[1, ]    NA    NA       NA
[2, ]    NA    NA       NA
[3, ]    NA    NA 3.066667
> cov(X, use="pairwise.complete.obs")
     [, 1]  [, 2]     [, 3]
[1, ]   4.0  4.50 3.500000
[2, ]   4.5 14.50 4.750000
[3, ]   3.5  4.75 3.066667
> cov(X, use="complete.obs")
         [, 1]     [, 2]     [, 3]
[1, ] 4.916667 4.500000 4.333333
[2, ] 4.500000 4.333333 4.333333
[3, ] 4.333333 4.333333 4.666667
> Y=cbind(c(1,2,4,6), c(2,4,5,7), c(-2,1,2,3)) #Eliminamos las
    filas con algún NA
> Y
     [, 1] [, 2] [, 3]
[1, ]     1     2   -2
[2, ]     2     4    1
[3, ]     4     5    2
[4, ]     6     7    3
> cov(Y)  #Dará lo mismo que con use="complete.obs"
         [, 1]     [, 2]     [, 3]
[1, ] 4.916667 4.500000 4.333333
[2, ] 4.500000 4.333333 4.333333
[3, ] 4.333333 4.333333 4.666667
```

Ejemplo 4.10. Recordaréis el *data frame* `iris`, que tabulaba las longitudes y anchuras de los pétalos y los sépalos de una muestra de flores iris de tres especies. Vamos a extraer un sub*data frame* con sus cuatro variables numéricas y calcularemos sus matrices de covarianzas y correlaciones.

```
 > str(iris)
'data.frame': 150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5
    ...
```

```
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1
   ...
 $ Species     : Factor w/ 3 levels "setosa", "versicolor", ..:
   1 1 1 1 1 1 1 1 1 1 ...
> iris_num=iris[, 1:4]
> cov(iris_num)  #Covarianzas muestrales
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    0.6856935  -0.0424340    1.2743154   0.5162707
Sepal.Width    -0.0424340   0.1899794   -0.3296564  -0.1216394
Petal.Length    1.2743154  -0.3296564    3.1162779   1.2956094
Petal.Width     0.5162707  -0.1216394    1.2956094   0.5810063
> n=dim(iris_num)[1]    #Número de filas; son 150, recordemos
> cov(iris_num)*(n-1)/n  #Covarianzas "verdaderas"
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length   0.68112222 -0.04215111    1.2658200   0.5128289
Sepal.Width   -0.04215111  0.18871289   -0.3274587  -0.1208284
Petal.Length   1.26582000 -0.32745867    3.0955027   1.2869720
Petal.Width    0.51282889 -0.12082844    1.2869720   0.5771329
> cor(iris_num)    #Correlaciones
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    1.0000000  -0.1175698    0.8717538   0.8179411
Sepal.Width    -0.1175698   1.0000000   -0.4284401  -0.3661259
Petal.Length    0.8717538  -0.4284401    1.0000000   0.9628654
Petal.Width     0.8179411  -0.3661259    0.9628654   1.0000000
```

Observamos, por ejemplo, una gran correlación lineal positiva entre la longitud y la anchura de los pétalos, 0.9628654, lo que indica una estrecha relación lineal con pendiente positiva entre estas magnitudes. Valdría la pena, entonces, calcular la recta de regresión lineal de una de estas medidas en función de la otra:

```
> lm(Petal.Length~Petal.Width, data=iris_num)

Call:
lm(formula = Petal.Length ~ Petal.Width, data = iris_num)

Coefficients:
(Intercept)  Petal.Width
      1.084        2.230

> summary(lm(Petal.Length~Petal.Width, data=iris_num))$r.squared
[1] 0.9271098
```

En cambio, la correlación entre la longitud y la anchura de los sépalos es -0.1175698, muy cercana a cero, lo que es señal de que la variación conjunta de las longitudes y anchuras de los sépalos no tiene una tendencia clara.

Vamos a ordenar ahora los pares de variables numéricas de `iris` en orden decreciente de su correlación en valor absoluto, para saber cuáles están más correlacionadas (en positivo o

negativo). Para ello, en primer lugar creamos un *data frame* cuyas filas están formadas por pares diferentes de variables numéricas de `iris`, su correlación y el valor absoluto de ésta, y a continuación ordenamos las filas de este *data frame* en orden decreciente de estos valores absolutos. Todo esto lo llevamos a cabo con el código siguiente, que luego explicamos:

```
> medidas=names(iris_num)
> n=length(medidas)  #En este caso, n=4
> indices=upper.tri(diag(n))
> medida1=matrix(rep(medidas, times=n), nrow=n,
  byrow=FALSE)[indices]
> medida2=matrix(rep(medidas, times=n), nrow=n,
  byrow=TRUE)[indices]
> corrs=as.vector(cor(iris_num))[indices]
> corrs.abs=abs(corrs)
> corrs_df=data.frame(medida1, medida2, corrs, corrs.abs)
> corrs_df
       medida1      medida2      corrs corrs.abs
1 Sepal.Length  Sepal.Width -0.1175698 0.1175698
2 Sepal.Length Petal.Length  0.8717538 0.8717538
3  Sepal.Width Petal.Length -0.4284401 0.4284401
4 Sepal.Length  Petal.Width  0.8179411 0.8179411
5  Sepal.Width  Petal.Width -0.3661259 0.3661259
6 Petal.Length  Petal.Width  0.9628654 0.9628654
> corrs_df_sort=corrs_df[order(corrs_df$corrs.abs,
   decreasing=TRUE), ]
> corrs_df_sort
       medida1      medida2      corrs corrs.abs
6 Petal.Length  Petal.Width  0.9628654 0.9628654
2 Sepal.Length Petal.Length  0.8717538 0.8717538
4 Sepal.Length  Petal.Width  0.8179411 0.8179411
3  Sepal.Width Petal.Length -0.4284401 0.4284401
5  Sepal.Width  Petal.Width -0.3661259 0.3661259
1 Sepal.Length  Sepal.Width -0.1175698 0.1175698
```

Vemos que el par de variables con mayor correlación en valor absoluto son `Petal.Length` y `Petal.Width`, como ya habíamos observado, seguidos por `Petal.Length` y `Sepal.Length`.

Vamos a explicar el código. La función `upper.tri`, aplicada a una matriz cuadrada M, produce la matriz *triangular superior* de valores lógicos del mismo orden que M, cuyas entradas (i, j) con $i < j$ son todas `TRUE` y el resto todas `FALSE`. Existe una función similar, `lower.tri`, para producir matrices *triangulares inferiores* de valores lógicos.

```
> upper.tri(diag(4))
      [,1]  [,2]  [,3]  [,4]
[1,] FALSE  TRUE  TRUE  TRUE
[2,] FALSE FALSE  TRUE  TRUE
[3,] FALSE FALSE FALSE  TRUE
[4,] FALSE FALSE FALSE FALSE
```

```
> lower.tri(diag(4))
      [,1]  [,2]  [,3]  [,4]
[1,] FALSE FALSE FALSE FALSE
[2,]  TRUE FALSE FALSE FALSE
[3,]  TRUE  TRUE FALSE FALSE
[4,]  TRUE  TRUE  TRUE FALSE
```

Ambas funciones disponen del parámetro `diag` que, igualado a `TRUE`, define también como `TRUE` las entradas de la diagonal principal.

```
> upper.tri(diag(4), diag=TRUE)
      [,1]  [,2]  [,3] [,4]
[1,]  TRUE  TRUE  TRUE TRUE
[2,] FALSE  TRUE  TRUE TRUE
[3,] FALSE FALSE  TRUE TRUE
[4,] FALSE FALSE FALSE TRUE
```

Si M es una matriz y L es una matriz de valores lógicos del mismo orden, `M[L]` produce el vector construido de la manera siguiente: de cada columna, se queda sólo con las entradas de M cuya entrada correspondiente en L es `TRUE`, y a continuación concatena estas columnas, de izquierda a derecha, en un vector.

```
> M=matrix(1:16, nrow=4, byrow=T)
> M
     [,1] [,2] [,3] [,4]
[1,]    1    2    3    4
[2,]    5    6    7    8
[3,]    9   10   11   12
[4,]   13   14   15   16
> M[upper.tri(diag(4))] #Las entradas del triángulo superior,
    por columnas
[1]  2  3  7  4  8 12
```

Ahora, tenemos las matrices siguientes:

```
> matrix(rep(medidas, times=4), nrow=4, byrow=FALSE)
     [,1]           [,2]           [,3]           [,4]
[1,] "Sepal.Length" "Sepal.Length" "Sepal.Length" "Sepal.Length"
[2,] "Sepal.Width"  "Sepal.Width"  "Sepal.Width"  "Sepal.Width"
[3,] "Petal.Length" "Petal.Length" "Petal.Length" "Petal.Length"
[4,] "Petal.Width"  "Petal.Width"  "Petal.Width"  "Petal.Width"
> matrix(rep(medidas, times=4), nrow=4, byrow=TRUE)
     [,1]           [,2]          [,3]           [,4]
[1,] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
[2,] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
[3,] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
[4,] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
```

Por lo tanto, al aplicar estas matrices a la matriz de valores lógicos `upper.tri(diag(4))` obtenemos los nombres de las variables correspondientes a las filas y las columnas del triángulo superior, respectivamente, y al aplicar la matriz de correlaciones a esta matriz de valores lógicos, obtenemos sus entradas en este triángulo; en los tres vectores, las entradas siguen el mismo orden. Esto nos permite construir el *data frame* `corrs_df` cuyas filas están formadas por pares diferentes de variables numéricas de `iris`, su correlación y, aplicando `abs` a esta última variable, su correlación en valor absoluto.

Finalmente, la función `order` ordena los valores del vector al que se aplica, en orden decreciente si se especifica el parámetro `decreasing=TRUE`. Cuando aplicamos un *data frame* a una de sus variables reordenada de esta manera, reordena sus filas según el orden de esta variable. En este caso hubiéramos conseguido lo mismo con la función `sort`, pero la función `order` se puede aplicar a más de una variable del *data frame*: esto permite ordenar las filas del *data frame* en el orden de la primera variable de manera que, en caso de empate, queden ordenadas por la segunda variable, y así sucesivamente.

4.5. Representación gráfica de datos multidimensionales

La representación gráfica de tablas de datos multidimensionales tiene la dificultad de las dimensiones; para dos o tres variables es sencillo visualizar las relaciones entre las mismas, pero para más variables ya no nos bastan nuestras tres dimensiones espaciales y tenemos que usar algunos trucos, tales como representaciones gráficas conjuntas de pares de variables.

Cuando tenemos una tabla de datos formada por dos variables numéricas, la manera más sencilla de representarlos gráficamente es mediante la función `plot` aplicada a la matriz de datos o al *data frame*. Con esta función obtenemos un gráfico de los puntos del plano que definen las filas de la tabla: en el contexto de la estadística multidimensional, se le llama el *diagrama de dispersión* (*scatter plot*) de los datos.

A modo de ejemplo, si extrajéramos de la tabla `iris` una subtabla conteniendo sólo las longitudes y anchuras de los pétalos y quisiéramos visualizar la relación entre estas dimensiones, podríamos dibujar su diagrama de dispersión de la manera siguiente:

```
> iris.pet=iris[ ,c("Petal.Length","Petal.Width")]
> plot(iris.pet, pch=20, xlab="Largo", ylab="Ancho")
```

El resultado es la Figura 4.1, que muestra una clara tendencia positiva: cuanto más largos son los pétalos, más anchos tienden a ser. Esto se corresponde con la correlación de 0.9628654 que hemos obtenido en el Ejemplo 4.10.

Para tablas de datos de tres columnas numéricas, podemos usar con un fin similar la instrucción `scatterplot3d` del paquete homónimo, que dibuja un diagrama de dispersión tridimensional. Como `plot`, se puede aplicar a un *data frame* o a una matriz; por ejemplo, para representar gráficamente las tres primeras variables numéricas de `iris`, podríamos hacer lo siguiente:

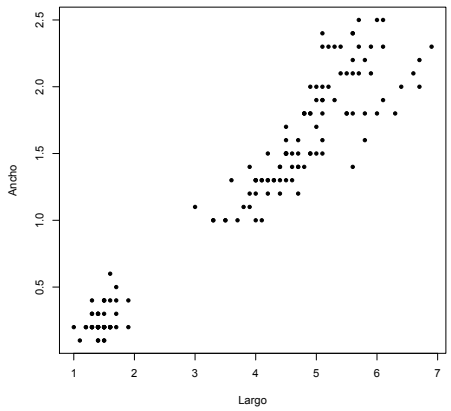


Figura 4.1. Diagrama de dispersión de las longitudes y anchuras de los pétalos de las flores representadas en la tabla `iris`.

```
> #Instalamos y cargamos el paquete scatterplot3d
...
> scatterplot3d(iris[ , 1:3], pch=20)
```

Obtendríamos la Figura 4.2. Podéis consultar el `help` de la instrucción para saber cómo modificar su apariencia: cómo ponerle un título, poner nombres adecuados a los ejes, usar colores, cambiar el estilo del gráfico, etc.

Una representación gráfica muy popular de las tablas de datos de tres o más columnas numéricas son las matrices formadas por los diagramas de dispersión de todos sus pares de columnas. Si la tabla de datos es un *data frame*, esta matriz de diagramas de dispersión se obtiene simplemente aplicando la función `plot` al *data frame*; por ejemplo,

```
> plot(iris[ , 1:4])
```

produce el gráfico de la izquierda de la Figura 4.3. En este gráfico, los cuadrados en la diagonal indican a qué variables corresponden cada fila y cada columna, de manera que podamos identificar fácilmente qué variables compara cada diagrama de dispersión; así, en el diagrama de la primera fila y segunda columna de esta figura, las abscisas corresponden a anchuras de sépalos y las ordenadas a longitudes de sépalos. Observad que la nube de puntos no muestra una tendencia clara y en todo caso ligeramente negativa, lo que se corresponde con la correlación entre estas variables de -0.11 que hemos obtenido en el Ejemplo 4.10.

Podemos usar los parámetros usuales de `plot` para mejorar el gráfico resultante; por ejemplo, podemos usar colores para distinguir las flores según su especie:

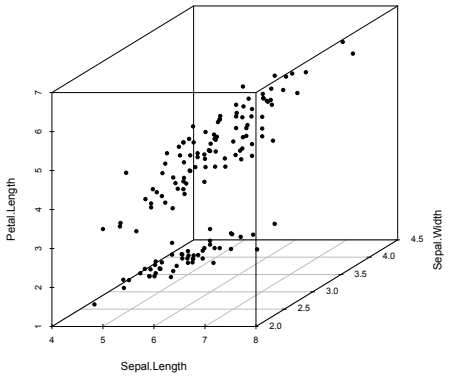


Figura 4.2. Diagrama de dispersión tridimensional de las tres primeras columnas de la tabla `iris`.

```
> plot(iris[ , 1:4], col=iris$Species)
```

produce el gráfico de la derecha de la Figura 4.3.

Para obtener la matriz de diagramas de dispersión de una tabla de datos multidimensional también se puede usar la función `pairs`: así, `pairs(iris[, 1:4])` produce exactamente el mismo gráfico que `plot(iris[, 1:4])`. La ventaja de `pairs` es que se puede aplicar a una matriz para obtener la matriz de diagramas de dispersión de sus columnas, mientras que `plot` no.

El paquete `car` incorpora una función que permite dibujar matrices de diagramas de dispersión enriquecidos con información descriptiva extra de las variables de la tabla de datos. Se trata de la función `spm`; por ejemplo,

```
> #Instalamos y cargamos el paquete car
...
> spm(iris[ , 1:4])
```

produce el gráfico de la izquierda de la Figura 4.4. Observaréis para empezar que en los cuadrados de la diagonal ha dibujado unas curvas: se trata de la curva de densidad de la variable correspondiente. La información gráfica contenida en estos cuadrados de la diagonal se puede modificar con el parámetro `diagonal`: podemos pedir, por ejemplo, que dibuje un histograma de cada variable (con `diagonal="histogram"`) o su *boxplot* (con `diagonal="boxplot"`). Así,

```
> spm(iris[ , 1:4], diagonal="boxplot")
```

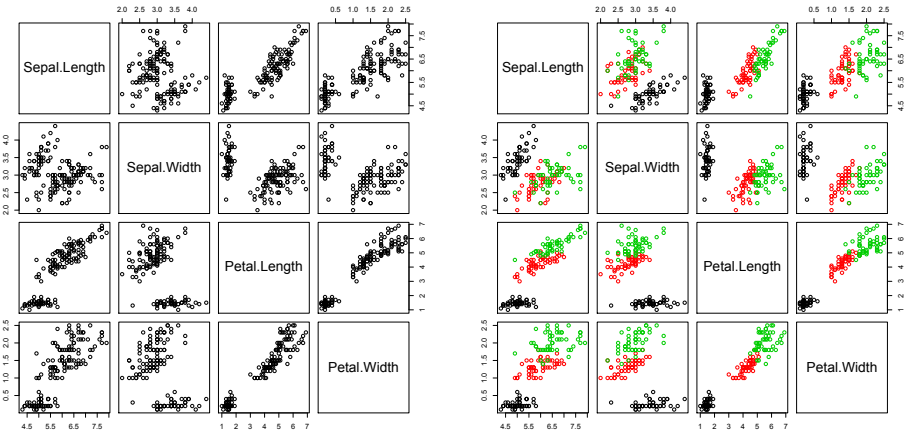


Figura 4.3. Matrices de diagramas de dispersión de la tabla `iris`; en la de la derecha, las especies se distinguen por medio de colores.

produce el gráfico de la derecha de la Figura 4.4.

Asimismo, observaréis que los diagramas de dispersión de la matriz producida con `spm` contienen curvas. La línea recta verde es la recta de regresión por mínimos cuadrados y, sin entrar en detalle sobre su significado exacto, las curvas rojas continuas representan la tendencia de los datos.

A veces querremos agrupar los datos de las variables numéricas de una tabla de datos. Los motivos serán los mismos que cuando se trata de una sola variable: por ejemplo, si los datos son aproximaciones de valores reales, o si son muy heterogéneos. Cuando tenemos dos variables emparejadas agrupadas, se pueden representar gráficamente las frecuencias de sus pares de clases mediante un *histograma bidimensional*, que divide el conjunto de todos los pares de valores en rectángulos definidos por los pares de intervalos e indica sobre cada rectángulo su frecuencia absoluta, por ejemplo mediante colores o intensidades de gris (dibujar barras verticales sobre las regiones es una mala idea, las de delante pueden ocultar las de detrás). Hay muchos paquetes de `R` que ofrecen funciones para dibujar histogramas bidimensionales; aquí explicaremos la función `hist2d` del paquete `gplots`. Su sintaxis básica es

`hist2d(`x`, `y`, nbins=..., col=...)`,

donde:

- x e y son los vectores de primeras y segundas coordenadas de los puntos. Si son las dos columnas de un *data frame* de dos variables, lo podemos entrar en su lugar.
- `nbins` sirve para indicar los números de clases: podemos igualarlo a un único valor,

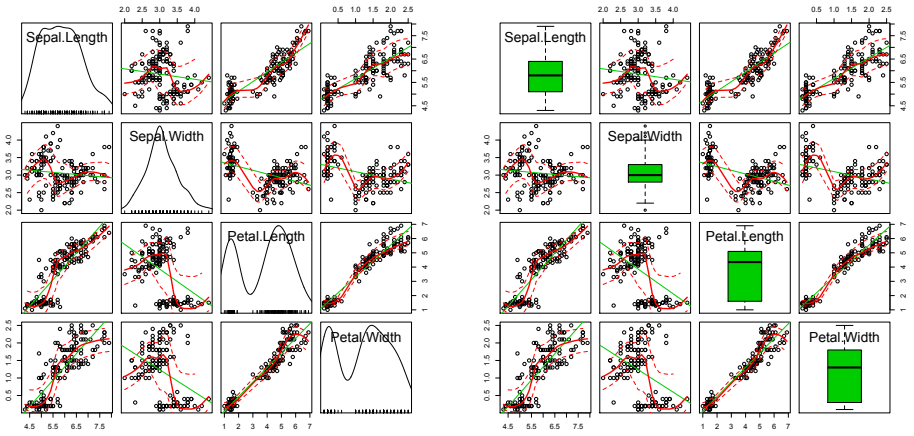


Figura 4.4. Matrices de diagramas de dispersión de la tabla `iris` producidos con `spm`.

y tomará ese número de clases sobre cada vector, o a un vector de dos entradas que indiquen el número de clases de cada vector.

- `col` sirve para especificar los colores a usar. Por defecto, los rectángulos vacíos aparecen de color negro, y el resto se colorean con tonalidades de rojo, de manera que los tonos más cálidos indican frecuencias mayores.

Además, podemos usar los parámetros usuales de `plot` para poner un título, etiquetar los ejes, etc.

A modo de ejemplo, vamos a dibujar el histograma bidimensional de las longitudes y anchuras de los pétalos de las flores iris, agrupando ambas dimensiones en los números de clases que da la regla de Freedman-Diaconis (y que calcula la función `nclass.FD`):

```
> #Instalamos y cargamos el paquete gplots
...
> hist2d(iris$Petal.Length, iris$Petal.Width,
  nbins=c(nclass.FD(iris$Petal.Length),
  nclass.FD(iris$Petal.Width)))
```

Al entrar esta última instrucción, obtenemos (junto con una serie de información en la consola que no hemos copiado) la Figura 4.5, que podéis comparar con el diagrama de dispersión de los mismos datos de la Figura 4.1.

En los histogramas bidimensionales con muchas regiones de diferentes frecuencias, es conveniente usar de manera adecuada los colores para representarlas. Una posibilidad es usar el paquete `RColorBrewer`, que permite elegir esquemas de colores bien diseñados. Las dos funciones básicas son:

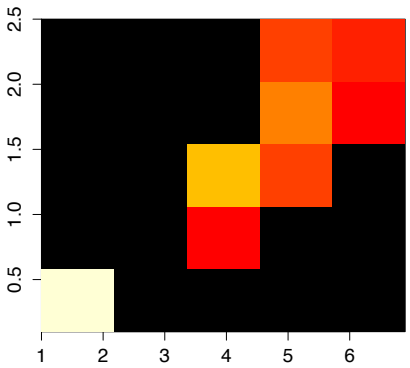


Figura 4.5. Histograma bidimensional de longitudes y anchuras de pétalos de flores iris.

- `brewer.pal(`n`, "`*paleta predefinida*`")`, que carga en un vector de colores (una *paleta*) una secuencia de n colores de la *paleta predefinida* en el paquete. Los nombres y contenidos de todas las paletas predefinidas que se pueden usar en esta función se obtienen, en la ventana de gráficos, ejecutando la instrucción `display.brewer.all()`. Por ejemplo, la paleta de colores de la Figura 4.6.(a) se define con el código:

```
> #Instalamos y cargamos el paquete RColorBrewer
...
> brewer.pal(11,"Spectral")
```

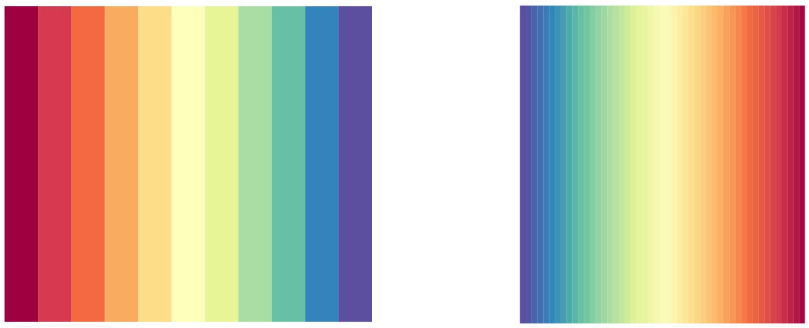
- `colorRampPalette(brewer.pal(...))(`m`)`, produce una nueva paleta de m colores a partir del resultado de `brewer.pal`, interpolando nuevos colores. Luego se puede usar la función `rev` para invertir el orden de los colores, lo que es conveniente en los histogramas bidimensionales si queremos que las frecuencias bajas correspondan a tonos azules y las frecuencias altas a tonos rojos. Así, la paleta de colores que se define con

```
> rev(colorRampPalette(brewer.pal(11,"Spectral"))(50))
```

  es la de la Figura 4.6.(b).

Vamos a usar esta última paleta en un histograma bidimensional de la tabla de alturas de padres e hijos recogidas por Karl Pearson en 1903 y que tenemos guardada en el *url* `http://aprender.uib.es/Rdir/pearson.txt`; el resultado es la Figura 4.7.

```
> df_pearson=read.table("http://aprender.uib.es/Rdir/pearson.txt
    ", header=TRUE)
```



(a) (b)

Figura 4.6. (a) Paleta `brewer.pal(11,"Spectral")`; (b) Paleta `rev(colorRampPalette(brewer.pal(11,"Spectral"))(50))`.

```
> hist2d(df_pearson, nbins=30,
    col=rev(colorRampPalette(brewer.pal(11,"Spectral"))(50)))
```

Para terminar, veamos como producir un gráfico conjunto de un histograma bidimensional y los dos histogramas unidimensionales.[3] Considerad la función siguiente, cuyos parámetros son un *data frame* `df` de dos variables y un número `n` de clases, común para las dos variables:

```
> hist.doble=function(df,n){
  par.anterior=par()
  h1=hist(df[,1], breaks=n, plot=F)
  h2=hist(df[,2], breaks=n, plot=F)
  m=max(h1$counts, h2$counts)
  par(mar=c(3,3,1,1))
  layout(matrix(c(2,0,1,3),nrow=2,byrow=T),
     heights=c(1,3), widths=c(3,1))
  hist2d(df, nbins=n,
     col=rev(colorRampPalette(brewer.pal(11,"Spectral"))(50)))
  par(mar=c(0,2,1,0))
  barplot(h1$counts, axes=F, ylim=c(0, m), col="red")
  par(mar=c(2,0,0.5,1))
  barplot(h2$counts, axes=F, xlim=c(0, m), col="red", horiz=T)
  par.anterior}
```

Entonces,

[3] Se trata de una modificación del gráfico similar explicado en `http://www.everydayanalytics.ca/2014/09/5-ways-to-do-2d-histograms-in-r.html`, el cual a su vez se inspira en un gráfico de la p. 62 de *Computational Actuarial Science with R* de Arthur Charpentier (Chapman and Hall/CRC, 2014).

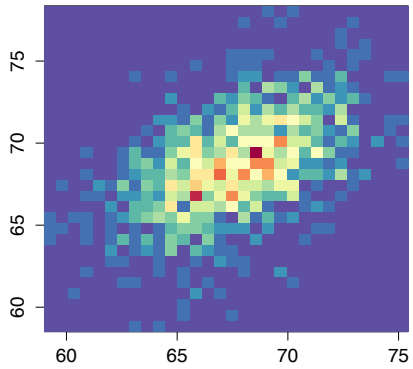


Figura 4.7. Histograma bidimensional de las alturas de padres e hijos recogidas por Karl Pearson.

```
> hist.doble(df_pearson,25)
```

produce la Figura 4.8.

Algunas explicaciones sobre el código, por si lo queréis modificar:

- Hemos «simulado» los histogramas mediante diagramas de barras de sus frecuencias absolutas, para poder dibujar horizontal el de la segunda variable.
- El parámetro `axes=FALSE` en los `barplot` indica que no dibuje sus ejes de coordenadas.
- La función `par` establece los parámetros generales básicos de los gráficos. Como con esta función los modificamos, guardamos los parámetros anteriores en `par.anterior` y al final los restauramos.
- El parámetro `mar` de la función `par` sirve para especificar, por este orden, los márgenes inferior, izquierdo, superior y derecho de la próxima figura, en números de líneas.
- La instrucción `layout` divide la figura a producir en sectores con la misma estructura que la matriz de su primer argumento. Dentro de esta matriz, cada entrada indica qué figura de las próximas se ha de situar en ese sector. Las alturas y amplitudes relativas de los sectores se especifican con los parámetros `heights` y `widths`, respectivamente. Así, la instrucción

```
layout(matrix(c(2,0,1,3),nrow=2,byrow=T), heights=c(1,3),
  widths=c(3,1))
```

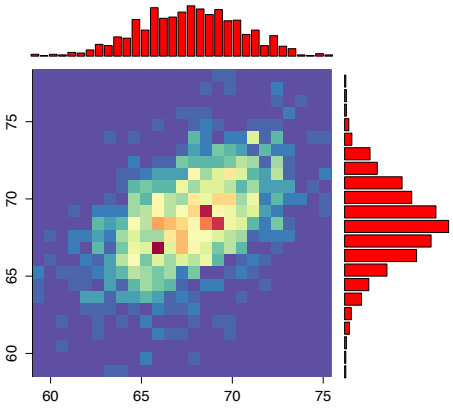


Figura 4.8. Histograma bidimensional con histogramas unidimensionales de las alturas de padres e hijos recogidas por Karl Pearson.

divide la figura en 4 sectores. Los sectores de la izquierda serán el triple de anchos que los de la derecha (`widths=c(3,1)`), y los sectores inferiores serán el triple de altos que los superiores (`heights=c(1,3)`). En estos sectores, `R` dibujará los próximos gráficos según el esquema definido por la matriz del argumento:

$$\begin{pmatrix} \text{segundo} & \text{ninguno} \\ \text{primero} & \text{tercero} \end{pmatrix}.$$

4.6. Guía rápida

- `sapply(`*data frame*`,`*función*`)` aplica la *función* a las columnas de un *data frame*.
- `scale` sirve para aplicar una transformación lineal a una matriz o a un *data frame*. Sus parámetros son:
  - `center`: especifica el vector que restamos a sus columnas.
  - `scale`: especifica el vector por el que dividimos sus columnas.
- `cov`, aplicada a dos vectores, calcula su covarianza muestral; aplicada a un *data frame* o a una matriz, calcula su matriz de covarianzas muestrales. Dispone del parámetro `use`:
  - Igualado a `"pairwise.complete.obs"`, calcula la covarianza de cada par de columnas teniendo en cuenta sólo sus observacions completas (las filas en las que ninguna de las dos tiene un `NA`), independientemente del resto de la tabla.

    - Igualado a `"complete.obs"`, calcula las covarianzas de las columnas teniendo en cuenta sólo las filas completas de toda la matriz.

- `cor`, aplicada a dos vectores, calcula su correlación; aplicada a un *data frame* o a una matriz, calcula su matriz de correlaciones. Se puede usar el parámetro `use` de `cov`.
- `cov2cor`, aplicada a la matriz de covarianzas, calcula la matriz de correlaciones.
- `upper.tri`, aplicada a una matriz cuadrada M, produce la matriz triangular superior de valores lógicos del mismo orden que M. Con el parámetro `diag=TRUE` se impone que el triángulo de valores `TRUE` incluya la diagonal principal.
- `lower.tri`, aplicada a una matriz cuadrada M, produce la matriz triangular inferior de valores lógicos del mismo orden que M. Dispone del mismo parámetro `diag=TRUE`.
- `order` ordena el primer vector al que se aplica, desempatando empates mediante el orden de los vectores subsiguientes a los que se aplica; el parámetro `decreasing=TRUE` sirve para especificar que sea en orden decreciente.
- `plot`, aplicado a un *data frame* de dos variables numéricas, dibuja su diagrama de dispersión; aplicado a un *data frame* de más de dos variables numéricas, produce la matriz formada por los diagramas de dispersión de todos sus pares de variables.
- `pairs` es equivalente a `plot` en el sentido anterior, y se puede aplicar a matrices.
- `scatterplot3d`, del paquete homónimo, dibuja diagramas de dispersión tridimensionales.
- `hist2d`, del paquete `gplots`, dibuja histogramas bidimensionales. Dispone de los parámetros específicos siguientes:
    - `nbins`: indica los números de clases.
    - `col`: especifica la paleta de colores que ha de usar para representar las frecuencias.
- `brewer.pal(`n`, "`*paleta predefinida*`")`, del paquete `RColorBrewer`, carga en una paleta de colores una secuencia de n colores de la *paleta predefinida* en dicho paquete.
- `colorRampPalette(brewer.pal(...))(`m`)`, del paquete `RColorBrewer`, produce una nueva paleta de m colores a partir del resultado de `brewer.pal`, interpolando nuevos colores.
- `display.brewer.all()`, del paquete `RColorBrewer`, muestra los nombres y contenidos de todas las paletas predefinidas en dicho paquete.
- `par` sirve para establecer los parámetros generales básicos de los gráficos.
- `layout` divide en sectores la figura a producir, para que pueda incluir varios gráficos independientes simultáneamente.