Contrastes para bondad de ajuste

Inferencia Estadística

21 de mayo de 2024

Objetivo:

- Determinar si los datos se ajustan a una distribución.
- La distribución puede
  - estar completamente especificada (H_0 simple);
  - depender de parámetros desconocidos (H_0 compuesta).

1. Contraste χ^2

- Para variables aleatorias discretas con número finito de valores.
- En otro caso, se podría agrupar en un número finito de clases.

1.1. Hipótesis nula simple $H_0 \colon X \equiv F_0$

- X variable aleatoria; $\vec{X} = (X_1, \ldots, X_n)$ muestra aleatoria simple suya.
- X toma valores C_1, …, C_k.
- $\forall\, i = 1, \ldots, k$, n_i = número de individuos en $\vec{X}$ iguales a C_i.
- $\forall\, i = 1, \ldots, k$, $p_i = \Pr[X = C_i]$
- El contraste es

$$\begin{array}{lll} H_0 \colon & \forall\, i \in \{1, \ldots, k\} & p_i = \Pr_{F_0}[X = C_i] = p_i^0 \\ H_1 \colon & \exists\, i \in \{1, \ldots, k\} & p_i \neq p_i^0 \end{array}$$

- La resolución se basará en comparar
  - las frecuencias absolutas observadas, n_i
  - las frecuencias absolutas esperadas, $E_i = np_i^0$

Hay dos procedimientos asintóticamente equivalentes para realizar la comparación.

1.1.1. Razón de verosimilitudes

- Verosimilitud:

$$L(n_1,\ldots,n_k,p_1,\ldots,p_k) = L(\vec{n},\vec{p}) = \frac{n!}{n_1!\ldots n_k!}\prod_{i=1}^{k} p_i^{n_i}$$

- Espacio paramétrico de dimensión $k-1$:

$$\Theta = \left\{ p_1,\ldots,p_{k-1} \;\middle|\; 0 < \sum_{i=1}^{k-1} p_i < 1 \right\}$$

- Logverosimilitud:

$$\begin{aligned}
\ln L(\vec{x},\vec{p}) &= \ln\left(\frac{n!}{n_1!\ldots n_k!}\right) + \sum_{i=1}^{k} n_i \ln p_i \\
&= \ln\left(\frac{n!}{n_1!\ldots n_k!}\right) + \sum_{i=1}^{k-1} n_i \ln p_i + n_k \ln p_k
\end{aligned}$$

- Ecuaciones de verosimilitud:

$$\begin{aligned}
\forall\, i \in \{1,\ldots,k-1\} \qquad 0 &= \frac{\partial \ln L(\vec{x},\vec{p})}{\partial p_i} \\
&= 0 + \frac{n_i}{p_i} + n_k \frac{\partial \ln p_k}{\partial p_i} \\
\langle p_k = 1 - p_1 - \cdots - p_{k-1} \rangle \quad &= \frac{n_i}{p_i} - \frac{n_k}{p_k} \\
&\Rightarrow \quad p_i = n_i \frac{p_k}{n_k} \quad \forall\, i \in \{1,\ldots,k-1\} \quad \langle * \rangle \\
&\Rightarrow \quad \sum_{i=1}^{k-1} p_i = \sum_{i=1}^{k-1} n_i \frac{p_k}{n_k} \\
&\Rightarrow \quad 1 - p_k = (n - n_k)\frac{p_k}{n_k} \\
&\Rightarrow \quad p_k = \frac{n_k}{n} \\
\langle * \rangle \quad &\Rightarrow \quad p_i = \frac{n_i}{n}
\end{aligned}$$

  luego $\forall\, i$, el estimador máximo-verosímil de p_i es $\hat{p}_i = \frac{n_i}{n}$.

- Estadígrafo de la razón de verosimilitudes:

$$\begin{aligned}
\Lambda &= \frac{L(\vec{n},\vec{p}\mid H_0)}{\sup\limits_{\vec{p}\in\Theta} L(\vec{n},\vec{p})} = \frac{\frac{n!}{n_1!\ldots n_k!}\prod_{i=1}^{k}(p_i^0)^{n_i}}{\frac{n!}{n_1!\ldots n_k!}\prod_{i=1}^{k}\hat{p}_i^{n_i}} = \prod_{i=1}^{k}\left(\frac{p_i^0}{n_i/n}\right)^{n_i} \\
&= \prod_{i=1}^{k}\left(\frac{E_i}{n_i}\right)^{n_i}
\end{aligned}$$

$$G = -2\ln\Lambda = 2\sum_{i=1}^{k} n_i \ln\frac{n_i}{E_i} \overset{H_0}{\equiv} \chi^2_{k-1} \quad \text{cuando } n \to \infty$$

$$\text{R.C.} = \{\vec{x} \mid G > c\}$$

1.1.2. Pearson

Se basa en las discrepancias cuadráticas entre las frecuencias observadas y las esperadas:

$$D = \sum_{i=1}^{k} \frac{(n_i - E_i)^2}{E_i} \quad \overset{H_0}{\equiv} \quad \chi^2_{k-1} \quad \text{cuando } n \to \infty$$

$$\text{R.C.} = \{\vec{x} \mid D > c\}$$

Es asintóticamente equivalente a G:

- $G = -2 \ln \Lambda = -2 \sum_{i=1}^{k} n_i \ln \frac{E_i}{n_i}$
- Por Taylor, dada una función f derivable, existe x' tal que $|x'| \leq |x|$ y

$$f(x) = f(0) + f'(0)\frac{x}{1!} + f''(0)\frac{x^2}{2!} + f'''(x')\frac{x^3}{3!}$$

- Tomando $f(x) = \ln(1+x)$, se tiene $f'(x) = \frac{1}{1+x}$, $f''(x) = \frac{-1}{(1+x)^2}$, $f'''(x) = \frac{2}{(1+x)^3}$.
- Por tanto, $\exists\, x'$ tal que $|x'| \leq |x|$ y

$$\ln(1+x) = 0 + x - \frac{1}{2}x^2 + \frac{2}{(1+x')^3}\frac{x^3}{3!} = x - \frac{1}{2}x^2 + \frac{x^3}{3(1+x')^3}$$

- $\exists\, x'_i$ tal que $|x'_i| \leq \left|\frac{n_i - E_i}{n_i}\right|$ y

$$\begin{aligned} \ln \frac{E_i}{n_i} &= \ln \frac{n_i - n_i + E_i}{n_i} = \ln\left(1 + \frac{-n_i + E_i}{n_i}\right) \\ &= \frac{-n_i + E_i}{n_i} - \frac{1}{2}\left(\frac{-n_i + E_i}{n_i}\right)^2 + \frac{1}{3(1+x'_i)^3}\left(\frac{-n_i + E_i}{n_i}\right)^3 \end{aligned}$$

- Combinando $\forall\, i$:

$$\begin{aligned} G &= -2\sum_{i=1}^{k} n_i \ln \frac{E_i}{n_i} \\ &= -2\sum_{i=1}^{k} n_i \left[\frac{-n_i + E_i}{n_i} - \frac{1}{2}\left(\frac{-n_i + E_i}{n_i}\right)^2 + \frac{1}{3(1+x'_i)^3}\left(\frac{-n_i + E_i}{n_i}\right)^3\right] \\ &= -2\sum_{i=1}^{k}(-n_i + E_i) + \sum_{i=1}^{k}\frac{(-n_i + E_i)^2}{n_i} - 2\sum_{i=1}^{k} n_i \frac{1}{3(1+x'_i)^3}\left(\frac{-n_i + E_i}{n_i}\right)^3 \end{aligned}$$

- $-2\sum_{i=1}^{k}(-n_i + E_i) = -2\sum(-n_i + np_i^0) = 2\left(-\sum n_i + n\sum p_i^0\right) =$
  $= 2(-n + n1) = 0$

- La versión de Pearson aparece al usar que $\hat{p}_i \xrightarrow[\text{c.s.}]{H_0} p_i^0$ en el segundo término:

$$\begin{aligned} \sum_{i=1}^{k}\frac{(-n_i + E_i)^2}{n_i} &= \sum_{i=1}^{k}\frac{(n_i - E_i)^2}{n_i} = \sum_{i=1}^{k}\frac{(n_i - E_i)^2 E_i/n}{E_i n_i/n} \\ &= \sum_{i=1}^{k}\frac{(n_i - E_i)^2 p_i^0}{E_i \hat{p}_i} \xrightarrow[\text{c.s.}]{H_0} \sum_{i=1}^{k}\frac{(n_i - E_i)^2}{E_i} = D \end{aligned}$$

- Respecto al residuo $-2\sum_{i=1}^{k} n_i \frac{1}{3(1+x_i')^3}\left(\frac{-n_i+E_i}{n_i}\right)^3$ de la expansión de Taylor, por un lado

$$\begin{aligned} n_i\left|\frac{n_i-E_i}{n_i}\right|^3 &= n_i\left|\frac{n_i-E_i}{n_i}\frac{n}{n}\right|^3 \\ &= n_i\left|\frac{n_i-E_i}{n}\frac{n}{n_i}\right|^3 \\ &= n_i\left|(\hat{p}_i-p_i^0)\frac{1}{\hat{p}_i}\right|^3 \\ &= \frac{n_i}{\hat{p}_i^3}\left|\hat{p}_i-p_i^0\right|^3 \\ &= \frac{n_i(\hat{p}_i-p_i^0)^2\left|\hat{p}_i-p_i^0\right|}{\hat{p}_i^3} \\ &= \frac{\left[\sqrt{n_i}(\hat{p}_i-p_i^0)\right]^2\left|\hat{p}_i-p_i^0\right|}{\hat{p}_i^3} \\ &\xrightarrow[\Pr]{H_0} 0 \end{aligned}$$

pues

$$\left.\begin{array}{lll} & \sqrt{n_i}(\hat{p}_i-p_i^0) \xrightarrow[\mathcal{L}]{H_0} N\left(0,\sqrt{p_i^0[1-p_i^0]}\right) & \\ \hat{p}_i-p_i^0 \xrightarrow[\text{c.s.}]{H_0} 0 & & \\ \hat{p}_i \xrightarrow[\text{c.s.}]{H_0} p_i^0>0 & & \end{array}\right\} \xRightarrow{\text{Mann\&Wald}} \frac{|\hat{p}_i-p_i^0|}{(\hat{p}_i)^3} \xrightarrow[\text{c.s.}]{H_0} 0 \left.\right\} \xRightarrow{\text{Slutsky}} \frac{\left[\sqrt{n_i}(\hat{p}_i-p_i^0)\right]^2\left|\hat{p}_i-p_i^0\right|}{\hat{p}_i^3} \xrightarrow[\Pr]{H_0} 0$$

- Por otro lado,

$$0 \le |x_i'| \le \left|\frac{n_i-E_i}{n_i}\right| \xrightarrow[\text{c.s.}]{H_0} 0$$

luego

$$\frac{1}{(1+x_i')^3} \xrightarrow[\text{c.s.}]{H_0} 1$$

y finalmente

$$-2\sum_{i=1}^{k} n_i \frac{1}{3(1+x_i')^3}\left(\frac{-n_i+E_i}{n_i}\right)^3 \xrightarrow[\text{c.s.}]{H_0} 0$$

1.1.3. Calidad de la aproximación asintótica

Para que sea aceptable, debe verificarse que

- $E_i \ge 5$ en al menos el 80 % de las clases C_i y
- $E_i \ge 1{,}5$ en el 20 % restante.

1.1.4. Ejemplo

Para comprobar si un dado hexagonal está perfectamente equilibrado, se lanza 120 veces y se obtienen las frecuencias

x_i	1	2	3	4	5	6
n_i	21	18	19	17	18	27

Contrasta si esos resultados se ajustan a una distribución con probabilidad $\frac{1}{6}$ por cara.

```
> O <- c(21,18,19,17,18,27) # observadas
> n <- sum(O)               # tamaño muestral
> E <- n * rep(1/6,6)       # esperadas
> E                         # todas > 5

[1] 20 20 20 20 20 20

> G <- 2 * sum (O * log (O / E)) # razón de verosimilitudes
> 1 - pchisq (G, length(O)-1)    # p-valor

[1] 0.6700914

> D <- sum ((O - E) ^ 2 / E)     # Pearson
> 1 - pchisq (D, length(O)-1)    # p-valor

[1] 0.6385699

> chisq.test (O)                 # el mismo p-valor

        Chi-squared test for given probabilities

data:  O
X-squared = 3.4, df = 5, p-value = 0.6386

> ### p-valor aproximado por Montecarlo
> ## usando chisq.test
> chisq.test (O, simulate.p.value=TRUE)

        Chi-squared test for given probabilities with simulated p-value (based
        on 2000 replicates)

data:  O
X-squared = 3.4, df = NA, p-value = 0.6672

> chisq.test (O, simulate.p.value=TRUE, B=10000)

        Chi-squared test for given probabilities with simulated p-value (based
        on 10000 replicates)

data:  O
X-squared = 3.4, df = NA, p-value = 0.6485
```

```
> ## "a mano"
> mean (replicate (10000,
+                  {
+                      O <- table (factor (sample(6, n, TRUE),
+                                          levels = 1:6))
+                      E <- n * rep(1/6,6)
+                      sum ((O-E)^2 / E)
+                  })
+       >= D)
```

```
[1] 0.6392
```

1.2. Hipótesis nula compuesta

En este caso,

$$H_0: \quad X \equiv F_\theta, \quad \theta \in \Theta \subset \mathbb{R}^q$$

o, teniendo en cuenta que sirve para variables discretas finitas,

$$H_0: \quad \Pr[X = C_i] = p_i(\theta), \quad \theta \in \Theta \subset \mathbb{R}^q$$

Se aplican los procedimientos anteriores pero

- sustituyendo el valor de θ por su estimación máximo-verosímil.
- teniendo en cuenta que la distribución asintótica es χ^2_{k-1-q}.

En el caso de agrupar los valores de la variable original para conseguir una distribución discreta finita, en rigor la EMV debería hacerse con los datos agrupados. En la práctica, la diferencia suele ser muy pequeña.

1.2.1. Ejemplo: Poisson

Considera la muestra con frecuencias

x_i	0	1	2	3	4	5	6
n_i	1	7	7	11	6	4	4

- Comprueba si puede proceder de una Poisson con $\lambda = 3$.
- Comprueba si puede proceder de una Poisson.

```
> ### landa = 3
>
> O <- c(1,7,7,11,6,4,4)
> n <- sum (O)
> p <- c (dpois(0:5,3), 1-ppois(5.5,3))
> E <- n * p
> E # alguna menor que 5
```

```
[1] 1.991483 5.974448 8.961672 8.961672 6.721254 4.032753 3.356718
```

```
> chisq.test (O, p = p, simulate.p.value = TRUE)

        Chi-squared test for given probabilities with simulated p-value (based
        on 2000 replicates)

data:  O
X-squared = 1.7636, df = NA, p-value = 0.949

> ### landa cualquiera
>
> ## EMV de landa con la muestra original
> weighted.mean (0:6, O)

[1] 3.05

> ## EMV de landa con la muestra agrupada
> vero <- function (landa) prod (dpois(0:5,landa) ^ O[1:6]) *
+                               (1-ppois(5,landa)) ^ O[7]
> optimize (vero, c(1,10), maximum=TRUE)

$maximum
[1] 3.113556

$objective
[1] 2.091637e-32

> ## una versión más robusta numéricamente
> logvero <- function (landa) sum (dpois(0:5,landa,log=TRUE) * O[1:6]) +
+                 ppois(5,landa,lower.tail=FALSE,log.p=TRUE) * O[7]
> optimize (logvero, c(1,10), maximum=TRUE)

$maximum
[1] 3.113566

$objective
[1] -72.94478

> E <- n * c (dpois(0:5,3.113566), 1-ppois(5.5,3.113566))
> E # cuidado si el p-valor queda cerca del nivel de significación

[1] 1.777688 5.534948 8.616713 8.942901 6.961078 4.334755 3.831917

> D <- sum ((O-E)^2/E)
> 1 - pchisq (D, length(O)-1-1) # se estima un parámetro

[1] 0.8926026
```

```
> ## agrupando categorías primera y segunda, y penúltima y última
> k <- length (O)
> O <- c (O[1]+O[2], O[-c(1,2,(k-1),k)], O[k-1]+O[k])
> logvero <- function (landa)
+     ppois (1, landa, log=TRUE) * O[1] +
+         sum (dpois(2:4,landa,log=TRUE) * O[2:4]) +
+         ppois(4, landa, lower.tail=FALSE, log.p=TRUE) * O[5]
> optimize (logvero, c(1,10), maximum=TRUE)

$maximum
[1] 3.076927

$objective
[1] -64.0125

> E <- n * c (ppois(1,3.076927), dpois(2:4,3.076927), 1-ppois(4,3.076927))
> E

[1] 7.517969 8.729150 8.952986 6.886921 7.912974

> D <- sum ((O-E)^2 / E)
> 1 - pchisq (D, length(O)-1-1) # se estima un parámetro

[1] 0.8117421
```

Como los distintos P-valores son muy altos, la muestra puede proceder de una distribución de Poisson.

1.2.2. Ejemplo: gausiana

¿Provienen los siguientes datos de una distribución gausiana?

(8.18, 7.81, 8.74, 11.95, 8.90, 9.33, 10.13, 11.23, 12.17, 9.76, 6.83, 10.79, 12.23, 8.82, 13.06, 6.34, 10.53, 10.39, 7.16, 10.59, 10.58, 10.74, 9.58, 11.11, 10.24, 10.20, 10.40, 7.14, 7.85, 11.65)

```
> x <- c(8.18,7.81,8.74,11.95,8.90,9.33,10.13,11.23,12.17,9.76,6.83,
+        10.79,12.23, 8.82,13.06,6.34,10.53,10.39,7.16,10.59,10.58,
+        10.74,9.58,11.11,10.24,10.20,10.40,7.14,7.85,11.65)
> n <- length (x)
> k <- floor (n / 5)
> (m <- mean (x))                          # EMV de mu

[1] 9.814333

> (s <- sd (x))                            # y sigma

[1] 1.733429
```

```
> u <- qnorm ((0:k)/k, m, s)              # umbrales
> O <- table (cut (x, u))                 # discretizar
> E <- n * diff (pnorm (u, m, s))
> rbind(O,E)

  (-Inf,8.14] (8.14,9.07] (9.07,9.81] (9.81,10.6] (10.6,11.5] (11.5, Inf]
O           6           4           3           6           6           5
E           5           5           5           5           5           5

> D <- sum ((O-E)^2/E)
> 1 - pchisq (D, k - 1 - 2) # dos parámetros estimados

[1] 0.6593898

> ## verosimilitud con datos agrupados
> logvero <- function (musigma, mu=musigma[1], sigma=musigma[2])
+     sum (O * log((pnorm (u[-1], mu, sigma)) -
+                  (pnorm (u[-(k+1)], mu, sigma))))
> (emv <- optim (c(m,s), logvero, control=list(fnscale=-1)) $ par)

[1] 9.843016 1.864933

> (E <- n * diff (pnorm (u, emv[1], emv[2])))

[1] 5.406141 4.757915 4.651877 4.680227 4.851979 5.651861

> rbind (O, E) # no se cumple por poco el criterio de convergencia

  (-Inf,8.14] (8.14,9.07] (9.07,9.81] (9.81,10.6] (10.6,11.5] (11.5, Inf]
O    6.000000    4.000000    3.000000    6.000000    6.000000    5.000000
E    5.406141    4.757915    4.651877    4.680227    4.851979    5.651861

> D <- sum ((O-E)^2 / E)
> 1 - pchisq (D, k-1 - 2) # dos parámetros estimados

[1] 0.684228
```

Sí puede ser gausiana. El P-valor se aleja mucho del nivel de significación, por lo que no hay duda. En este caso no se debe usar `simulate.p.value=TRUE` porque supondría admitir que las estimaciones son los verdaderos parámetros.*

*Se podrían muestrear los parámetros, pero eso exigiría suponer una distribución a priori sobre los mismos (inferencia bayesiana).

2. Prueba de Kolmogórov y Smirnov

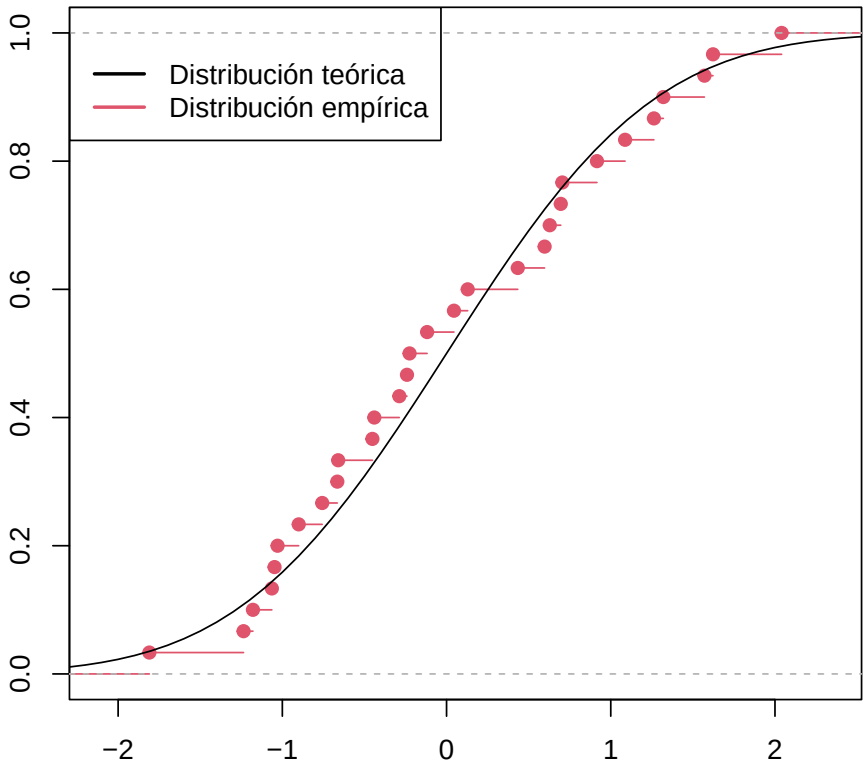
Es un contraste de bondad de ajuste a una distribuciónn continua completamente determinada. Se basa en la función de distribución empírica F_n y sus propiedades como estimador de la función de distribución teórica. Dadas una muestra aleatoria simple $(X_1, \ldots, X_n)$ de una variable aleatoria continua X y una hipótesis simple sobre el comportamiento de esa variable, que suponemos determinado por una función de distribución F_0, se plantean las hipótesis $H_0 \colon X \equiv F_0$, $H_1 \colon X \not\equiv F_0$ y se considera el estadístico:

$$D_n = \sup_{x \in \mathbb{R}} |F_n(x) - F_0(x)|$$

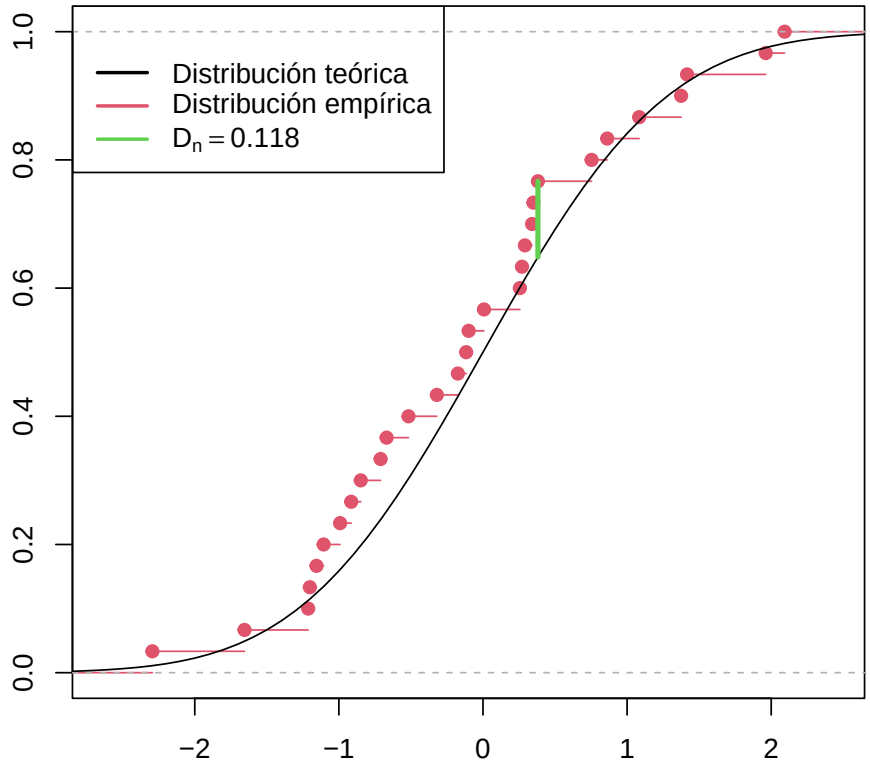
que mide la distancia entre la función de distribución empírica y la teórica. Se rechaza la hipótesis nula cuando su valor es alto:

$$\mathrm{RC} = [D_n > c]$$

```
> X <- rnorm (30)
> plot (ecdf (X), col=2, main="", xlab="", ylab="")
> curve (pnorm, -5, 5, add=TRUE)
> legend ("topleft",
+         c("","Distribución teórica","Distribución empírica"),
+         col = 0:2, lwd = 2)
```



```
> distro <- "norm" # "unif" "exp"
> F <- get (paste0 ("p", distro))
> n <- 30
> X <- sort (get (paste0 ("r", distro)) (n))
> D1 <- (1:n)/n - F(X);      D1e <- max(D1)
> D2 <- F(X) - (0:(n-1))/n; D2e <- max(D2)
> if (D1e > D2e) {e <- 0; Dn <- D1e; i <- which.max(D1)} else
+                {e <- 1; Dn <- D2e; i <- which.max(D2)}
> plot (ecdf (X), col=2, main="", xlab="", ylab="")
> curve (pnorm, -5, 5, add=TRUE)
> lines (rep(X[i],2), c(F(X[i]),(i-e)/n), col=3, lwd=3)
> legend ("topleft",
+         legend = c ("", "Distribución teórica", "Distribución empírica",
+                     eval (substitute (expression (D[n]==Dn),
+                                       list (Dn = round (Dn, 3))))),
+         col = 0:3, lwd = 2)
```



D_n toma valores entre 0 y 1 y, por el teorema de Glivenko y Cantelli, tiende a 0 cuando n aumenta. Para

estudiar su comportamiento se definen

$$D_n^+ = \sup_{x\in\mathbb{R}} F_n(x) - F_0(x)$$

$$D_n^- = \sup_{x\in\mathbb{R}} F_0(x) - F_n(x)$$

ya que

$$\begin{aligned} D_n &= \sup_{x\in\mathbb{R}} |F_n(x) - F_0(x)| = \sup_{x\in\mathbb{R}} \text{máx}\left\{F_n(x) - F_0(x), F_0(x) - F_n(x)\right\} \\ &= \text{máx}\left\{\sup_{x\in\mathbb{R}}[F_n(x) - F_0(x)], \sup_{x\in\mathbb{R}}[F_0(x) - F_n(x)]\right\} = \text{máx}\{D_n^+, D_n^-\} \end{aligned}$$

Su cálculo puede simplificarse teniendo en cuenta que la función de distribución empírica es constante en cada intervalo de la siguiente partición:

$$\mathbb{R} = (\{x_{(0)} \coloneqq -\infty\}, x_{(1)}) \cup [x_{(1)}, x_{(2)}) \cup \cdots \cup [x_{(n-1)}, x_{(n)}) \cup [x_{(n)}, \infty)$$

luego

$$D_n^+ = \max_{i=1}^{n} \frac{i}{n} - F_0(x_{(i)})$$

$$D_n^- = \max_{i=1}^{n} F_0(x_{(i)}) - \frac{i-1}{n}$$

Por tanto,

$$D_n(\vec{X}) = \text{máx}\left\{\frac{i}{n} - F_0(X_{(i)}), F_0(X_{(i)}) - \frac{i-1}{n} \;\middle|\; i = 1, \ldots, n\right\}$$

y, teniendo en cuenta que

$$F_0(X) \overset{H_0}{\equiv} U(0,1)$$

se tiene que $F_0(X_{(i)})$ se distribuye como el i-ésimo estadístico de orden $U_{(i)}$ de una muestra de tamaño n de una $U(0,1)$ y, conjuntamente $\left(F_0(X_{(n)}), \ldots, F_0(X_{(n)})\right) \overset{H_0}{\equiv} (U_{(1)}, \ldots, U_{(n)})$. Así, bajo H_0, D_n es de libre distribución, es decir, su distribución no depende de la distribución de X. Esto permite calcular la distribución de D_n bajo H_0 usando la distribución continua más sencilla, la $U(0,1)$:

$$\begin{aligned} D_n &= \sup_{x\in\mathbb{R}} |F_n(x) - F_0(x)| = \sup_{x\in\mathbb{R}} \left|\frac{\#\{X_i \le x\}}{n} - F_0(x)\right| = \sup_{x\in\mathbb{R}} \left|\frac{\#\{F_0(X_i) \le F_0(x)\}}{n} - F_0(x)\right| \\ &= \sup_{0\le u\le 1} \left|\frac{\#\{U_i \le u\}}{n} - u\right| = \text{máx}\left\{\frac{i}{n} - U_{(i)}, U_{(i)} - \frac{i-1}{n} \;\middle|\; i = 1, \ldots, n\right\} \end{aligned}$$

Puesto que los saltos que tiene la función de distribución empírica son múltiplos[†] de $\frac{1}{n}$, el valor más pequeño que puede tomar D_n es $\frac{1}{2n}$, que corresponde al mejor ajuste que se puede lograr con n saltos de altura $\frac{1}{n}$ en valores del intervalo $[0,1]$ a la función de distribución de la $U(0,1)$. Así, bajo H_0, la función de distribución

[†]De hecho, como se trata de variables continuas, la probabilidad de que se repita un valor y se obtenga, por tanto, un salto mayor que $\frac{1}{n}$, es nula.

de D_n es

$$\begin{aligned}
\Pr(D_n \le z) &= \Pr\left[\text{máx}\left\{\frac{i}{n} - U_{(i)}, U_{(i)} - \frac{i-1}{n} \;\middle|\; i = 1, \ldots, n\right\} \le z\right] \\
&= \Pr\left[\frac{i}{n} - U_{(i)}, U_{(i)} - \frac{i-1}{n} \le z \;\forall\, i = 1, \ldots, n\right] \\
&= \Pr\left[\frac{i}{n} - z \le U_{(i)} \le \frac{i-1}{n} + z \;\forall\, i = 1, \ldots, n\right] \\
\Pr\left[D_n \le \frac{1}{2n} + v\right] &= \begin{cases} 0 & v < 0 \\ \langle * \rangle & 0 \le v \le \frac{2n-1}{2n} \\ 1 & v > \frac{2n-1}{2n} \end{cases} \\
\langle * \rangle &= \underbrace{\int_{u_{(n)}=\frac{2n-1}{2n}-v}^{u_{(n)}=\frac{2n-1}{2n}+v} \cdots \int_{u_{(2)}=\frac{3}{2n}-v}^{u_{(2)}=\frac{3}{2n}+v} \int_{u_{(1)}=\frac{1}{2n}-v}^{u_{(1)}=\frac{1}{2n}+v}}_{0<u_{(1)}<\cdots<u_{(n)}<1} n!\, du_{(1)}\, du_{(2)} \ldots du_{(n)}
\end{aligned}$$

Es tedioso evaluar la integral porque el recinto de integración cambia de forma según el valor de v. Por ejemplo [3, pág. 108], para $n = 2$ y $0 < v < \frac{2n-1}{2n} = \frac{3}{4}$,

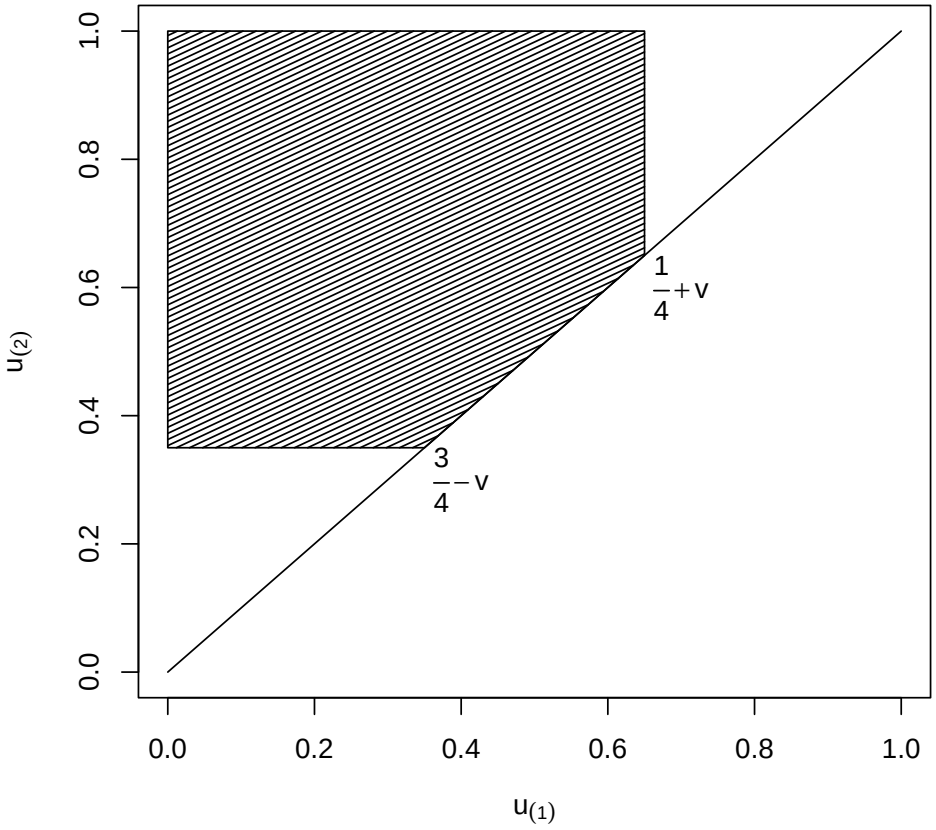
$$\Pr\left(D_2 \le \frac{1}{4} + v\right) = 2! \underbrace{\int_{u_{(2)}=\frac{3}{4}-v}^{u_{(2)}=\frac{3}{4}+v} \int_{u_{(1)}=\frac{1}{4}-v}^{u_{(1)}=\frac{1}{4}+v}}_{0<u_{(1)}<u_{(2)}<1} du_{(1)}\, du_{(2)}$$

No hay solapamiento entre los intervalos cuando $\frac{1}{4} + v > \frac{3}{4} - v$, es decir, cuando $v > \frac{1}{4}$. Luego, si $v < \frac{1}{4}$,

$$\Pr\left(D_2 \le \frac{1}{4} + v\right) = 2 \int_{u_{(2)}=\frac{3}{4}-v}^{u_{(2)}=\frac{3}{4}+v} \int_{u_{(1)}=\frac{1}{4}-v}^{u_{(1)}=\frac{1}{4}+v} du_{(1)}\, du_{(2)} = 2(2v)^2$$

Cuando $\frac{1}{4} < v < \frac{3}{4}$, sin embargo, el cálculo se complica porque el recinto de integración tiene la forma mostrada en la figura:

```
> plot (c(0,1), c(0,1), type="l",
+       xlab=expression(u[(1)]), ylab=expression(u[(2)]))
> v <- 0.4 # por ejemplo; 1/4 < v < 3/4
> polygon (c(    0,3/4-v,1/4+v,1/4+v,0),
+          c(3/4-v,3/4-v,1/4+v,    1,1),
+          30, 25)
> eps <- 0.05
> text(c(3/4-v,1/4+v)+eps, c(3/4-v,1/4+v)-eps,
+      c (expression(over(3,4)-v), expression(over(1,4)+v)))
```



Para muestras de tamaño grande se usa la aproximación

$$\Pr\left(D_n \leq \frac{z}{\sqrt{n}}\right) \xrightarrow[n\to\infty]{} 1 - \sum_{i=1}^{\infty}(-1)^{i-1}e^{-2i^2z^2}$$

La distribución se halla tabulada [1, 2] e implementada en entornos como R (`ks.test`). Alternativamente, es fácil implementar su distribución mediante Montecarlo usando $U(0,1)$.

La distribución de D_n sirve para determinar bandas de confianza en torno a la función de distribución teórica. Considerando el cuantil de D_n de orden $1-\alpha$, $d_{1-\alpha}$, se tiene

$$\begin{aligned} 1-\alpha &= \Pr[|F_n(x) - F_0(x)| \leq d_{1-\alpha}\ \forall\, x] \\ &= \Pr[-d_{1-\alpha} \leq F_n(x) - F_0(x) \leq d_{1-\alpha}\ \forall\, x] \\ &= \Pr[F_n(x) - d_{1-\alpha} \leq F_0(x) \leq F_n(x) + d_{1-\alpha}\ \forall\, x] \end{aligned}$$

luego las bandas serían

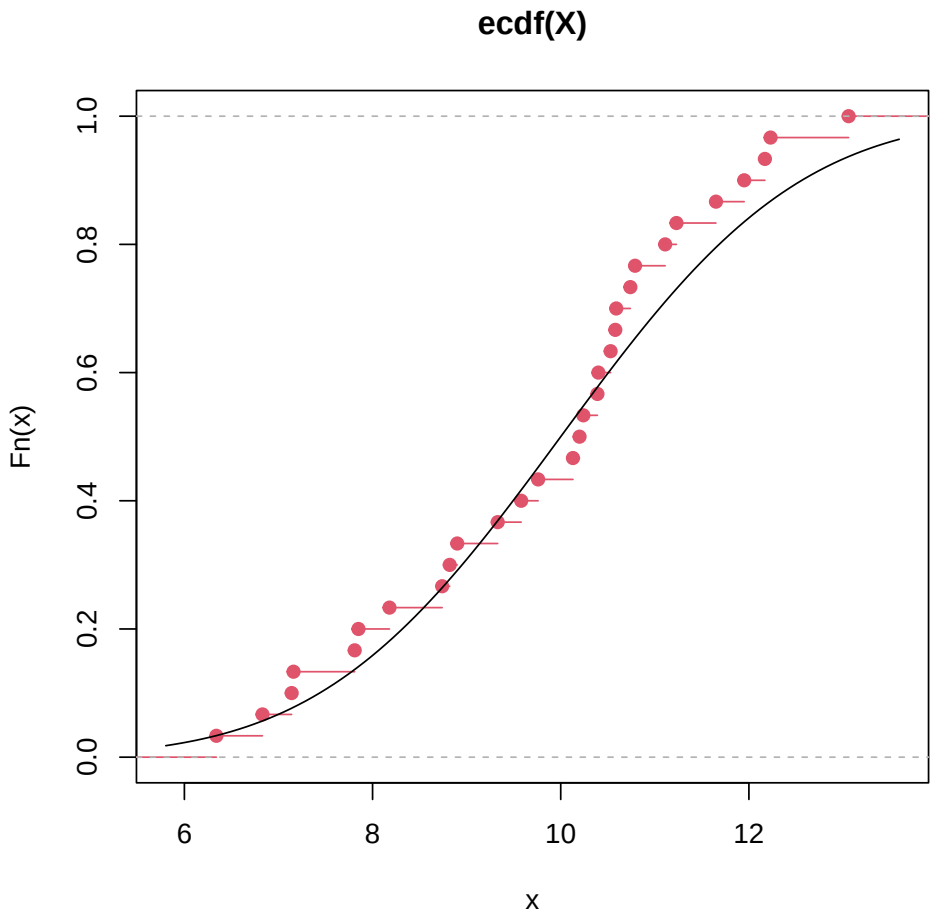
$$(\text{máx}\{0, F_n(x) - d_{1-\alpha}\}, \text{mín}\{1, d_{1-\alpha} - F_n(x)\}) \quad x \in \mathbb{R}$$

Para contrastes unilaterales se usan D_n^+ y D_n^-.

2.1. Ejemplo

Contrasta si los siguientes datos proceden de una distribución gausiana de media 10 y desvío típico 2: (8.18, 7.81, 8.74, 11.95, 8.90, 9.33, 10.13, 11.23, 12.17, 9.76, 6.83, 10.79, 12.23, 8.82, 13.06, 6.34, 10.53, 10.39, 7.16, 10.59, 10.58, 10.74, 9.58, 11.11, 10.24, 10.20, 10.40, 7.14, 7.85, 11.65).

```
> X <- c (8.18, 7.81, 8.74, 11.95, 8.90, 9.33, 10.13, 11.23, 12.17,
+         9.76, 6.83, 10.79, 12.23, 8.82, 13.06, 6.34, 10.53, 10.39,
+         7.16, 10.59, 10.58, 10.74, 9.58, 11.11, 10.24, 10.20,
+         10.40, 7.14, 7.85, 11.65)
> ### compararación gráfica
> plot (ecdf (X), col=2)                  # empírica
> F0 <- function (x) pnorm (x, 10, 2)     # teórica
> curve (F0, add=TRUE)
```




```
> ### función ks.test
> ks.test (X, F0)

        One-sample Kolmogorov-Smirnov test
```

```
data:  X
D = 0.11309, p-value = 0.797
alternative hypothesis: two-sided

> ks.test (X, pnorm, mean=10, sd=2) # lo mismo

        One-sample Kolmogorov-Smirnov test

data:  X
D = 0.11309, p-value = 0.797
alternative hypothesis: two-sided

> ### "a mano"
> Dn <- function (X)
+ {
+     n <- length (X)
+     Xo <- sort (X)
+     Dmas <- (1:n)/n - F0(Xo)
+     Dmenos <- F0(Xo) - (0:(n-1))/n
+     max (Dmas, Dmenos)
+ }
> Dn(X) # en la muestra del enunciado

[1] 0.1130881

> ## aproximación del Pvalor por Montecarlo
> mean (replicate (10000,
+                  Dn (rnorm (length(X), 10, 2)))
+       >= Dn(X))

[1] 0.7985
```

Los nombres de las opciones (`mean`, `sd`) relativas a los parámetros de las distribuciones se pueden consultar en la ayuda de R: `?pnorm`

Téngase en cuenta que, al ser D_n de libre distribución, la simulación de Montecarlo puede hacerse sin recurrir a la F_0 concreta:

```
> distriDn <- function (n)
+     replicate (10000,
+     {
+         Xo <- sort (runif (n))
+         Dmas <- (1:n)/n - Xo
+         Dmenos <- Xo - (0:(n-1))/n
+         max (Dmas, Dmenos)
+     })
> ## aproximación del Pvalor por Montecarlo
> mean (distriDn (length (X)) >= Dn(X))

[1] 0.7967
```

De hecho, es más cómodo usar directamente la función de R:

```
> n <- length (X)
> distriDn <- replicate (10000, ks.test (runif (n), punif) $ statistic)
> ## aproximación del Pvalor por Montecarlo
> mean (distriDn >= Dn(X)) # Dn(X) = sup|Fn-F0| con F0=N(10;2)

[1] 0.8023

> ks.test (X, pnorm, 10, 2) $ p.value

[1] 0.7970045

> ## región crítica RC = [Dn > c] con Pr[RC]=0,95
> quantile (distriDn, 0.95)

      95%
0.2397588
```

2.2. Ejemplo

El contraste de Kolmogórov y Smirnov se debe usar sólo cuando la distribución bajo H_0 está completamente especificada. Si no, sobreajusta la distribución y tiende a no rechazar la hipótesi nula.

```
> set.seed (2)
> X <- runif (200)
> ks.test (X, pnorm, mean = 1/2, sd = sqrt(1/12))

        One-sample Kolmogorov-Smirnov test

data:  X
D = 0.11822, p-value = 0.007469
alternative hypothesis: two-sided

> ks.test (X, pnorm, mean = mean(X), sd = sd(X))

        One-sample Kolmogorov-Smirnov test

data:  X
D = 0.09281, p-value = 0.06377
alternative hypothesis: two-sided
```

2.3. Caso discreto

En este caso D_n no es de libre distribución, pero es fácil obtener un P-valor mediante montecarlo o con una biblioteca informática:

```
> ### datos de ejemplo
> n <- 100 # tamaño muestral
> m <- 5; p0 <- .3 # parámetros de la distri bajo H0
> x <- rbinom (n, m, p0)
> ### estadístico
> F0 <- stepfun(0:m, pbinom(-1:m, m, p0))
> Dn <- function (x) max (abs (ecdf(x)(0:m) - F0(0:m)))
> ### montecarlo
> distri <- replicate (1e4, Dn(rbinom (n, m, p0)))
> mean (distri >= Dn(x))

[1] 0.8448

> ### biblioteca
> ## sudo apt-get install libfftw3-dev
> ## install.packages("KSgeneral")
> KSgeneral::disc_ks_test(x, F0, exact=TRUE)

        One-sample Kolmogorov-Smirnov test

data:  x
D = 0.02922, p-value = 0.8427
alternative hypothesis: two-sided
```

3. Prueba de Cramér y Von Mises

Es similar a la de Kolmogórov y Smirnov (que usa la distancia del supremo[‡]) pero para medir la distancia entre teórica y empírica usa la norma L^2:

$$W^2 = \int_{-\infty}^{\infty} (F_n(x) - F_0(x))^2 \, dF_0(x)$$

$$\mathrm{RC} = [W^2 > c]$$

Otra forma de expresarlo:[§]

$$\begin{aligned} nW^2 &= n\sum_{i=0}^{n} \int_{x_{(i)}}^{x_{(i+1)}} \left(\frac{i}{n} - F_0(x)\right)^2 dF_0(x) = n\sum_{i=0}^{n} \int_{F_0(x_{(i)})}^{F_0(x_{(i+1)})} \left(\frac{i}{n} - u\right)^2 du \\ &= \frac{1}{12n} + \sum_{i=1}^{n} \left(\frac{2i-1}{2n} - F_0(x_{(i)})\right)^2 = \frac{1}{12n} + \sum_{i=1}^{n} \left(\frac{2i-1}{2n} - u_{(i)}\right)^2 \end{aligned}$$

donde se ve que es de libre distribución. Para comprobar la igualdad en Maxima:

```
/* F[i] := F0(x(i)) */
n*sum(integrate((i/n-u)^2, u, F[i], F[i+1]), i, 0, n) ; /* Origen */
1/(12*n) + sum(((2*i-1)/(2*n)-F[i])^2, i, 1, n) ; /* Destino */
```

$$n\sum_{i=0}^{n} \frac{F_{i+1}^3\, n^2 - 3i\, F_{i+1}^2 n + 3i^2\, F_{i+1}}{3n^2} - \frac{F_i^3\, n^2 - 3i\, F_i^2 n + 3i^2\, F_i}{3n^2} \qquad (\%\ \mathrm{o1})$$

[‡]También llamada del infinito o de Chebichof (Chebyshëv, Чебышёв).

[§]Con $x_{(0)} = -\infty$ y $x_{(n+1)} = \infty$.

$$\frac{1}{12n} + \sum_{i=1}^{n} \left(\frac{2i-1}{2n} - F_i \right)^2 \qquad (\%\ o2)$$

```
(%i3) define (sumandoO(i), integrate((i/n-u)^2, u, F[i], F[i+1])) $
sumandoFiO :
n*subst ([F[i-1] = 0, F[i+1] = 0], ratsimp(sumandoO(i-1) + sumandoO(i))) ;
/* sumando asociado a F[i] */
define (sumandoD(i), ((2*i-1)/(2*n)-F[i])^2) $
sumandoFiD : expand (sumandoD(i)) $
sinFi(x) := freeof (F[i], x) $
conFi(x) := not sinFi(x) $
filtrar (suma, predicado) := apply ("+", sublist (maplist (identity, suma), predicado)) $
sumandoConFi : filtrar (sumandoFiD, conFi) ;
is (factor(sumandoFiO) = factor(sumandoConFi)) ;
/* los sumandos en F[i] coinciden */
```

$$\frac{3F_i^2 n + (3-6i)\ F_i}{3n} \qquad \text{(sumandoFiO)}$$

$$-\frac{2i\,F_i}{n} + \frac{F_i}{n} + F_i^2 \qquad \text{(sumandoConFi)}$$

$$\text{true} \qquad (\%\ o11)$$

```
(%i12) define (sumandoSinFi(i), filtrar (sumandoFiD, sinFi)) $
TIdestino : ratsimp (nusum (sumandoSinFi(i), i, 1, n) + 1/(12*n));
/* término independiente en Destino */
F[0] : 0 $F[n+1] : 1 $sinF(x) := freeof (F, x) $
TIorigen : filtrar (expand (n*sumandoO(n)), sinF) ;
is (TIorigen = TIdestino) ; /* los términos independientes coinciden */
```

$$\frac{n}{3} \qquad \text{(TIdestino)}$$

$$\frac{n}{3} \qquad \text{(TIorigen)}$$

$$\text{true} \qquad (\%\ o18)$$

```
> x # muestra del ejemplo anterior

  [1] 3 2 2 4 1 2 1 1 1 1 4 1 2 2 4 0 1 0 1 1 0 3 1 1 2 3 3 1 2 2 2 1 3 2 3 1 2
 [38] 1 1 1 3 1 2 2 3 3 1 1 0 1 2 0 1 0 2 0 0 1 0 2 2 0 2 1 2 2 0 2 3 2 3 3 3 1
 [75] 2 1 4 1 0 1 1 4 2 2 1 4 2 2 2 1 1 1 2 1 2 1 1 0 0 1

> stem (x) # representar distribución
```

```
  The decimal point is at the |

  0 | 00000000000000
  0 |
  1 | 0000000000000000000000000000000000000000
  1 |
  2 | 00000000000000000000000000000000
  2 |
  3 | 0000000000000
  3 |
  4 | 000000
```

```
> goftest::cvm.test (x, pnorm, mean=10, sd=2)

        Cramer-von Mises test of goodness-of-fit
        Null hypothesis: Normal distribution
        with parameters mean = 10, sd = 2
        Parameters assumed to be fixed

data:  x
omega2 = 33.311, p-value < 2.2e-16
```

```
> goftest::cvm.test (x, pnorm, mean=10, sd=2, estimated=TRUE)

        Cramer-von Mises test of goodness-of-fit
        Braun's adjustment using 10 groups
        Null hypothesis: Normal distribution
        with parameters mean = 10, sd = 2
        Parameters assumed to have been estimated from data

data:  x
omega2max = 3.3332, p-value < 2.2e-16
```

```
> ## con parámetros estimados, el p-valor puede cambiar
> goftest::cvm.test (x, pnorm, mean=10, sd=2, estimated=TRUE)

        Cramer-von Mises test of goodness-of-fit
        Braun's adjustment using 10 groups
        Null hypothesis: Normal distribution
        with parameters mean = 10, sd = 2
        Parameters assumed to have been estimated from data

data:  x
omega2max = 3.3331, p-value < 2.2e-16
```

De la ayuda `?goftest::cvm.test`

> Note that Braun's method involves randomly dividing the data into two equally-sized subsets, so the p-value is not exactly the same if the test is repeated. This technique is expected to work well when the number of observations in 'x' is large.

4. Prueba de Anderson y Darling

Se trata de otra prueba para distribuciones continuas. Comparada con KS y CvM, da más peso a las colas de la distribución:

$$A = n \int_{-\infty}^{\infty} \frac{[F_n(x) - F_0(x)]^2}{F_0(x)[1 - F_0(x)]}\, dF_0(x)$$

Al usar, como en las otras pruebas, estadísticos de orden trasformados por la función de distribución, se llega a la siguiente expresión en que se observa claramente que es un estadístico de libre distribución:

$$A = -n - \sum_{i=1}^{n} \frac{2i-1}{n} \left(\ln F_0(x_{(i)}) \ln[1 - F_0(x_{(n-i+1)})] \right)$$

```
/* origen: n*sum(integrate((i/n-u)^2/(u*(1-u)), u, F[i], F[i+1]), i, 0, n) */
/* destino: -n - sum((2*i-1)/n*(log(F[i])+log(1-F[n-i+1])), i, 1, n)       */
remarray(F) $
assume (F[i]>0, F[i+1]>0, F[i]<1, F[i+1]<1, F[i+1]>F[i]) $ /*la segunda sobra*/
define (sumandoO(i), ratsimp(integrate((i/n-u)^2/(u*(1-u)), u, F[i], F[i+1]))) $
define (sumandoD(i), (2*i-1)/n*(log(F[i])+log(1-F[n-i+1]))) $
/* no vale la anterior porque busco los tErminos con F[i] */
define (sumandoD(i),
  ratsimp((2*i-1)/n*log(F[i]) + (2*(n-i+1)-1)/n*log(1-F[i]))) $
/* origen tiene tErminos con F[i] para i y para i-1 */
sumandoOconFi(i) := partition(expand(sumandoO(i)),F[i])[2] +
                    partition(expand(sumandoO(i-1)),F[i])[2] $ /* para i=1..n */
is (expand(-sumandoD(i)) = expand(n*sumandoOconFi(i))) ; /* true */
sumando0 : partition(sumandoO(0),F[0])[2] $
F[0] : 0 $
''sumando0 ; /* 0 */
sumandoNmas1 : partition(expand(sumandoO(n)),F[n+1])[2] $
F[n+1] : 1 $
''sumandoNmas1 ; /* -1 */
/* para i=0..n los tErminos en F[i] coinciden             */
/* para i=0 el sumando en origen es nulo                  */
/* para i=n+1 el sumando en origen aporta -1 que, por n,  */
/*   es -n = el tErmino independiente del destino         */
```

```
> x # muestra del ejemplo anterior

  [1] 3 2 2 4 1 2 1 1 1 1 4 1 2 2 4 0 1 0 1 1 0 3 1 1 2 3 3 1 2 2 2 1 3 2 3 1 2
 [38] 1 1 1 3 1 2 2 3 3 1 1 0 1 2 0 1 0 2 0 0 1 0 2 2 0 2 1 2 2 0 2 3 2 3 3 3 1
 [75] 2 1 4 1 0 1 1 4 2 2 1 4 2 2 2 1 1 1 2 1 2 1 1 0 0 1

> goftest::ad.test (x, pnorm, mean=10, sd=2)

        Anderson-Darling test of goodness-of-fit
        Null hypothesis: Normal distribution
        with parameters mean = 10, sd = 2
        Parameters assumed to be fixed

data:  x
An = 909.88, p-value = 6e-06
```

```
> goftest::ad.test (x, pnorm, mean=10, sd=2, estimated=TRUE)

        Anderson-Darling test of goodness-of-fit
        Braun's adjustment using 10 groups
        Null hypothesis: Normal distribution
        with parameters mean = 10, sd = 2
        Parameters assumed to have been estimated from data

data:  x
Anmax = 101.64, p-value = 0.0005998

> goftest::ad.test (x, pnorm, mean=10, sd=2, estimated=TRUE) # cambia

        Anderson-Darling test of goodness-of-fit
        Braun's adjustment using 10 groups
        Null hypothesis: Normal distribution
        with parameters mean = 10, sd = 2
        Parameters assumed to have been estimated from data

data:  x
Anmax = 104.79, p-value = 0.0005998
```

5. Diagramas cuantil-cuantil y probabilidad-probabilidad

5.1. QQplot

Una gráfica cuantil-cuantil representa puntos cuyas coordenadas vienen dadas por cuantiles $\frac{i}{n}$ $(i = 1, ..., n)$:

- teóricos $F_X^{-1}\left(\frac{i}{n}\right)$ como abscisas;
- muestrales $x_{(i)}$ como ordenadas.

Como el cuantil de orden 1 es infinito para muchas distribuciones importantes (gausiana, exponencial, t, F...) se modifican los órdenes $\frac{i}{n}$ usando alguna corrección. Por ejemplo, la función `ppoints` de `R` usa

$$\frac{i-a}{n+(1-a)-a}$$

- Por omisión se usa el criterio de Blom [4]

$$a = \begin{cases} 3/8 & n \leqslant 10 \\ 1/2 & n \geqslant 11 \end{cases}$$

  que produce los cuantiles $\frac{i-\frac{3}{8}}{n+\frac{1}{4}}$ para muestras pequeñas.

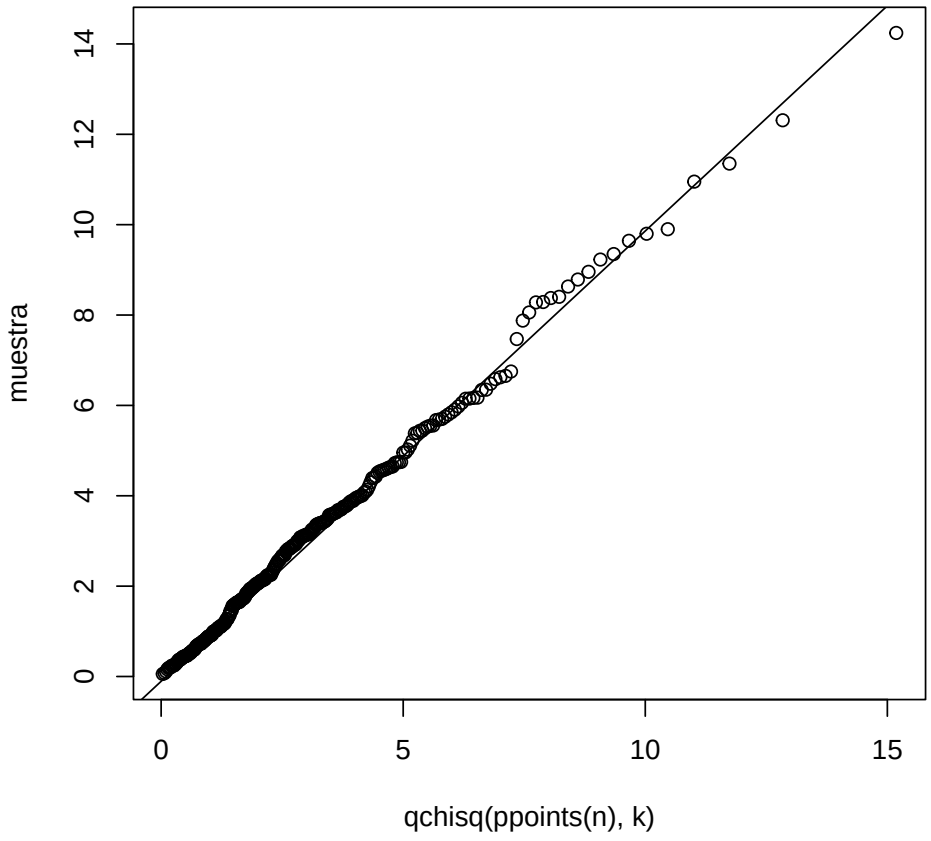
- Con $a = \frac{1}{3}$ se llega al criterio de Tukey: $\frac{i-\frac{1}{3}}{n+\frac{1}{3}}$
- Con $a = 0$ se tiene la corrección simple $\frac{i}{n+1}$

Los órdenes de tales cuantiles son siempre simétricos respecto a $\frac{1}{2}$:

```
(%i6) corregido(i) := (i-a) / (n + (1-a) -a) ;
                                              i - a
(%o6)                    corregido(i) := ---------------
                                         n + (1 - a) - a
(%i7) corregido(i) + corregido(n-i+1);
                             i - a      n - i - a + 1
(%o7)                      ---------- + -------------
                           n - 2 a + 1    n - 2 a + 1
(%i8) ratsimp(%);
(%o8)                                  1
```

Ejemplo en R para una χ^2_3:

```
> n <- 300; k <- 3
> muestra <- rchisq(n, k)
> qqplot (qchisq(ppoints(n), k), muestra)
> ## Para añadir una línea de referencia:
> qqline (muestra, distribution = function(x) qchisq(x,3))
```



5.2. Caso gausiano (QQnorm)

Si $X \equiv \mathcal{N}(\mu, \sigma)$, entonces su función de distribución es

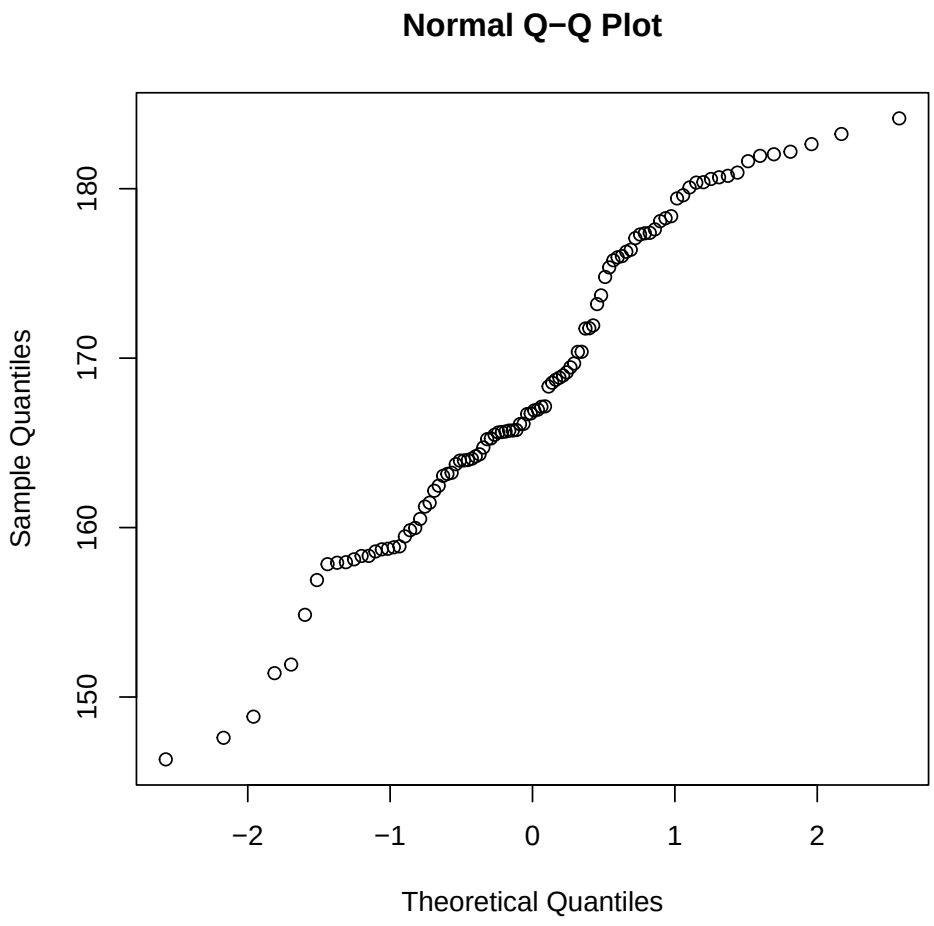
$$F_X(x) = \Phi\left(\frac{x-\mu}{\sigma}\right)$$

donde Φ es la de la gausiana típica $\mathcal{N}(0,1)$. En R se representan los puntos:

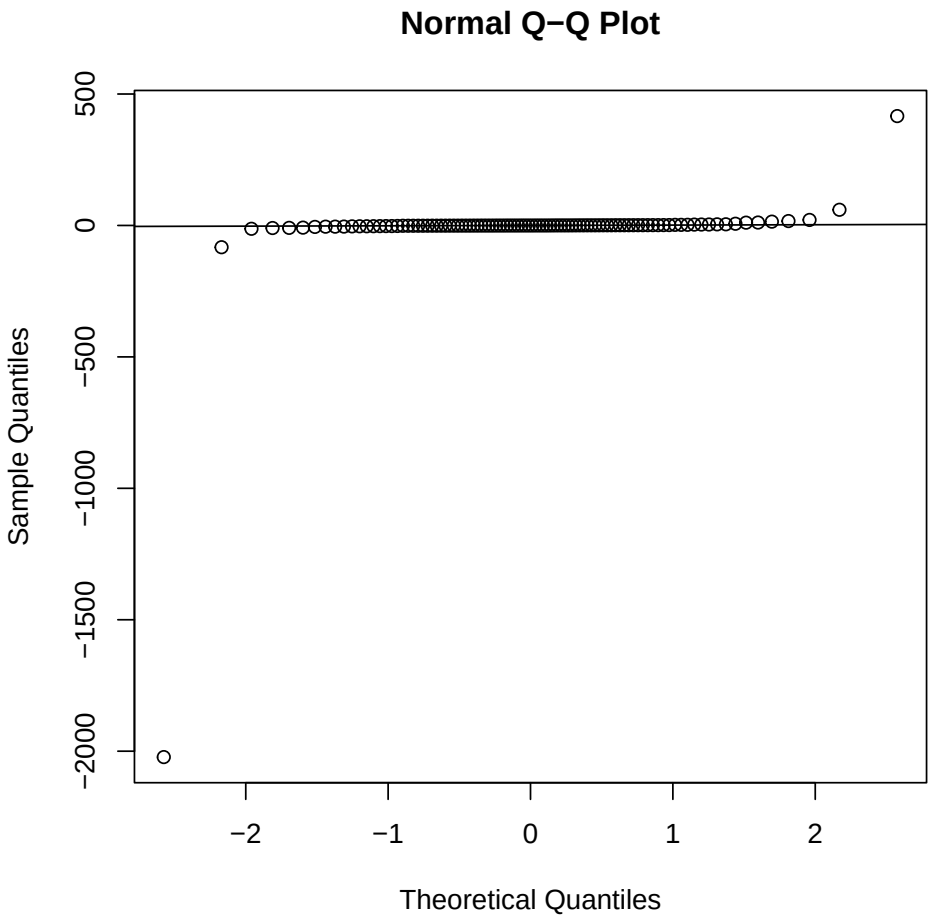
$$\left(\Phi^{-1}\left(\frac{i-a}{n+(1-a)-a}\right), x_{(i)}\right) \quad i = 1, \ldots, n$$

Si es cierta la hipótesis nula de gausianidad los puntos deberían disponerse en torno a una línea recta:

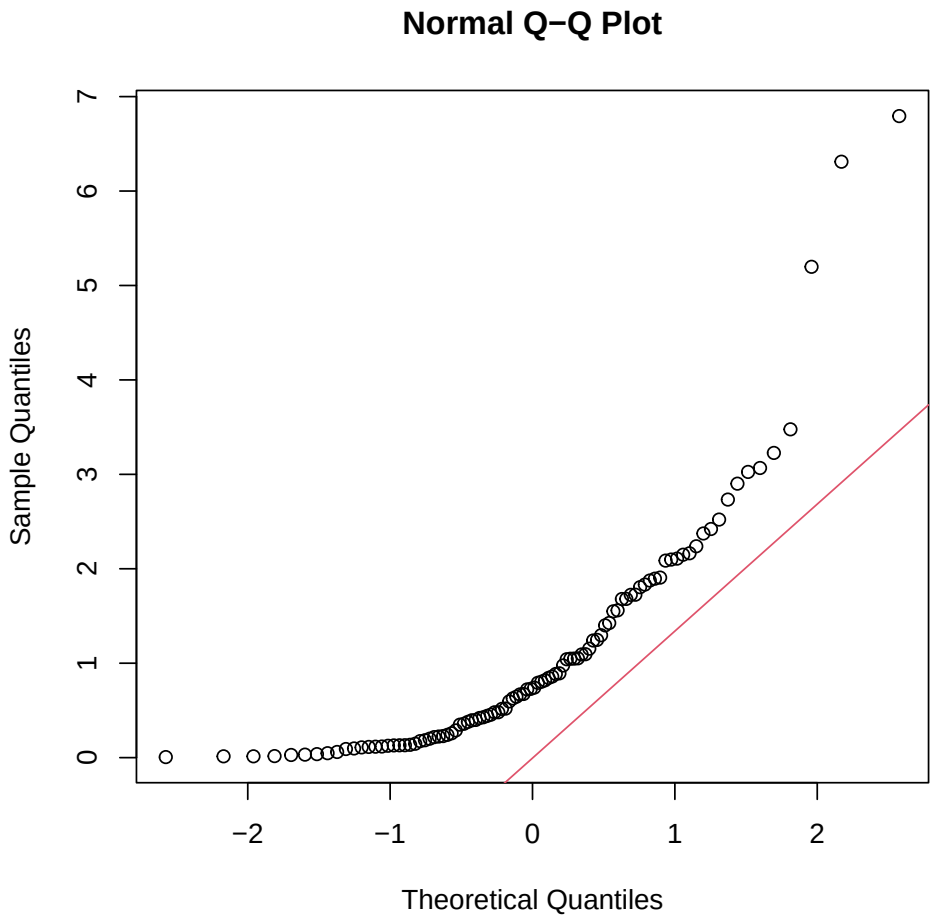
```
> n <- 100
> qqnorm (rnorm (n, 170, 10))
```




```
> qqnorm (x <- rt (n, 1)); qqline (x)
```




```
> qqnorm (rexp (n)); qqline (x, col=2)
```



5.3. PPplot

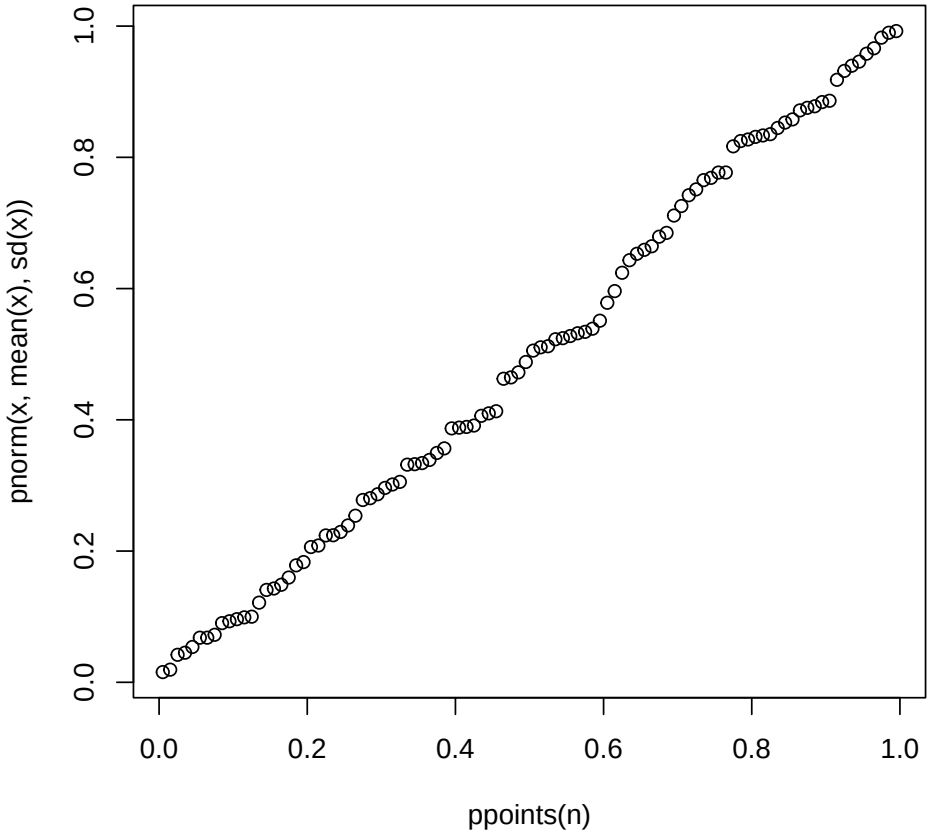
Una alternativa mucho menos usada son los diagramas probabilidad-probabilidad o porcentaje-porcentaje, que representan los puntos

$$\left(\frac{i-a}{n+(1-a)-a}, F_0(x_{(i)})\right) \quad i = 1, \ldots, n$$

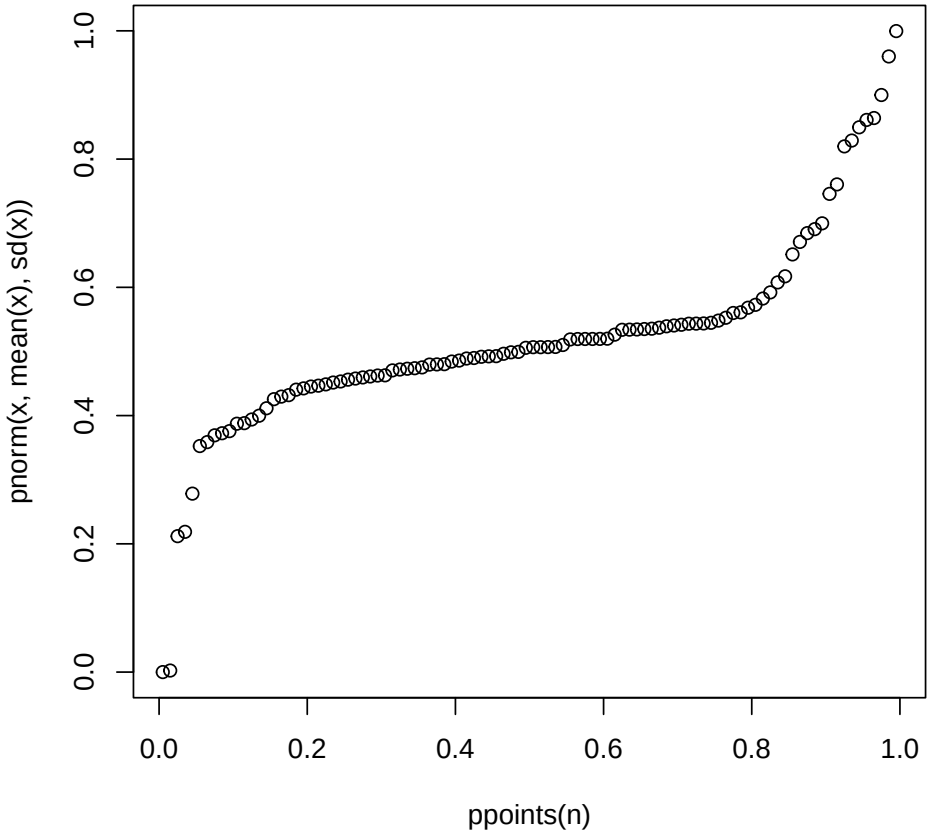
En el caso gausiano,

$$\left(\frac{i-a}{n+(1-a)-a}, \Phi\left(\frac{x_{(i)}-\bar{x}}{s}\right)\right) \quad i = 1, \ldots, n$$

```
> x <- sort (rnorm (n, 170, 10))
> plot (ppoints(n), pnorm(x,mean(x),sd(x)))
```




```
> x <- sort (rt (n, 1))
> plot (ppoints(n), pnorm(x,mean(x),sd(x)))
```



6. Pruebas de gausianidad

6.1. Lilliefors

Basada en una modificación del estadístico KS:

$$\begin{aligned} D_n &= \sup_{x\in\mathbb{R}} |F_n(x) - \hat{F}_0(x)| = \sup_{x\in\mathbb{R}} |F_n(x) - F_{\mathcal{N}(\bar{X},S)}x)| \\ &= \sup_{x\in\mathbb{R}} \left| F_n(x) - \Phi\left(\frac{x-\bar{X}}{S}\right)\right| \end{aligned}$$

Al usar en el ajuste los parámetros estimados a partir de la muestra, en general D_n tomará valores más pequeños que los de KS.

Al igual que la uniforme o la exponencial, la distribución gausiana mantiene su forma (salvo localización y

escala) por lo que D_n es de libre distribución bajo H_0: $X \equiv \mathcal{N}$.

$$\begin{aligned} D_n &= \sup_{x\in\mathbb{R}} \left| F_n(x) - \Phi\left(\frac{x-\bar{X}}{S_X}\right)\right| = \sup_{x\in\mathbb{R}} \left| \frac{\#\{X_i \leqslant x\}}{n} - \Phi\left(\frac{x-\bar{X}}{S_X}\right)\right| \\ &= \sup_{x\in\mathbb{R}} \left| \frac{\#\left\{\frac{X_i-\mu}{\sigma} \leqslant \frac{x-\mu}{\sigma}\right\}}{n} - \Phi\left(\frac{\frac{x-\mu}{\sigma} - \frac{\bar{X}-\mu}{\sigma}}{\frac{S_X}{\sigma}}\right)\right| \\ &= \sup_{y\in\mathbb{R}} \left| \frac{\#\{Y_i \leqslant y\}}{n} - \Phi\left(\frac{y-\bar{Y}}{S_Y}\right)\right| \end{aligned}$$

con $Y_i = \frac{X_i-\mu}{\sigma} \equiv \mathcal{N}(0,1)$. Por tanto, se puede simular su comportamiento generando muestras $\mathcal{N}(0,1)$.

```
> X <- 1:200
> nortest::lillie.test(X) # usando biblioteca de R

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  X
D = 0.059128, p-value = 0.08578

> ## P-valor aproximado por Montecarlo:
> n <- length(X)
> mean (replicate (1e4,
+                  ks.test (Xb <- rnorm(n),
+                           pnorm, mean(Xb), sd(Xb)) $ statistic)
+       >= ks.test (X, pnorm, mean(X), sd(X)) $ statistic)

[1] 0.0868
```

Puede emplearse (véase 34) esta estrategia también para la exponencial $\mathcal{E}(\lambda)$ y la uniforme $\mathcal{U}(0,\theta)$.

6.2. Shapiro y Francia

Se basa en la misma idea que las gráficas cuantil-cuantil. Considérese una muestra gausiana típica y sea $\vec{Z}_{(\cdot)} = (Z_{(1)}, \ldots, Z_{(n)})^t$ la correspondiente muestra ordenada. Sea $\vec{X}_{(\cdot)} = (X_{(1)}, \ldots, X_{(n)})^t$ una muestra ordenada proveniente de una población X. Si X es gausiana $\mathcal{N}(\mu,\sigma)$, entonces $\vec{X}_{(\cdot)}$ se puede expresar como trasformación lineal de $\vec{Z}_{(\cdot)}$:

$$X_{(i)} = \mu + \sigma Z_{(i)}$$

Entonces una medida adecuada de la gausianidad sería el coeficiente de determinación W entre $\vec{X}_{(\cdot)}$ y $\vec{m} = E(\vec{Z}_{(\cdot)}) \approx \Phi^{-1}\left(\frac{i-\frac{3}{8}}{n+\frac{1}{4}}\right)$:

$$R^2(\vec{x}_{(\cdot)}, \vec{m}) = \frac{\mathrm{Cov}^2(\vec{x}_{(\cdot)}, \vec{m})}{\mathrm{Var}(\vec{x}_{(\cdot)})\mathrm{Var}(\vec{m})} = \frac{\left[\sum (x_{(i)} - \bar{x})(m_i - \bar{m})\right]^2}{\sum (x_{(i)} - \bar{x})^2 \sum (m_i - \bar{m})^2} = \frac{\left[\sum m_i (x_{(i)} - \bar{x})\right]^2}{\sum (x_{(i)} - \bar{x})^2 \sum m_i^2}$$

La región crítica sería RC $= [W < c]$.

```
> X <- 1:60
> nortest::sf.test (X)
```

```
        Shapiro-Francia normality test

data:  X
W = 0.96619, p-value = 0.0872

> n <- length(X)
> Z <- qnorm (ppoints (n, a=3/8))
> W <- function (X) cor (sort(X), Z) ^ 2
> W(X) # igual que sf.test

[1] 0.9661861

> mean (replicate (1e4, W(rnorm(n))) <= W(X)) # P-valor aprox

[1] 0.0928
```

6.3. Shapiro y Wilk

Se trata de una prueba muy potente, por lo que se ha popularizado como la prueba de gausianidad por antonomasia[¶]:

```
> X <- 1:60
> shapiro.test (X)

        Shapiro-Wilk normality test

data:  X
W = 0.95523, p-value = 0.02761
```

Su idea es similar al de Shapiro y Francia pero tiene en cuenta también las covarianzas entre los estadígrafos de orden. Considérese una muestra gausiana típica y sea $\vec{Z}_{(\cdot)} = (Z_{(1)}, \ldots, Z_{(n)})^t$ la correspondiente muestra ordenada. Sean sus esperanzas y covarianzas

$$\vec{m} = E\begin{pmatrix} Z_{(1)} \\ \vdots \\ Z_{(n)} \end{pmatrix} \quad V = \mathrm{Var}\begin{pmatrix} Z_{(1)} \\ \vdots \\ Z_{(n)} \end{pmatrix} \quad Z_i \equiv \mathcal{N}(0,1)\ \forall\, i = 1, \ldots, n$$

Sea $\vec{X}_{(\cdot)} = (X_{(1)}, \ldots, X_{(n)})^t$ una muestra ordenada proveniente de una población X. Si X es gausiana $\mathcal{N}(\mu, \sigma)$, entonces se puede expresar

$$X_{(i)} = \mu + \sigma Z_{(i)}$$

Estimando μ y σ por mínimos cuadrados generalizados[‖] hay que minimizar los residuos "mahalanobizados"[**]

$$[\vec{x}_{(\cdot)} - \mu\vec{1} - \sigma\vec{z}_{(\cdot)}]^t V^{-1} [\vec{x}_{(\cdot)} - \mu\vec{1} - \sigma\vec{z}_{(\cdot)}] =$$

$$= \left(\vec{x}_{(\cdot)} - \begin{bmatrix}\vec{1} & \vec{z}_{(\cdot)}\end{bmatrix}\begin{bmatrix}\mu \\ \sigma\end{bmatrix}\right)^t V^{-1} \left(\vec{x}_{(\cdot)} - \begin{bmatrix}\vec{1} & \vec{z}_{(\cdot)}\end{bmatrix}\begin{bmatrix}\mu \\ \sigma\end{bmatrix}\right) =$$

[¶] Es la única disponible en R básico.

[‖] Si $\vec{y} = X\vec{\beta} + \vec{\epsilon}$ con $E(\vec{\epsilon}) = \vec{0}$ y $\mathrm{Var}(\epsilon) = V$, entonces $\hat{\beta} = (X^t V^{-1} X)^t X^t V^{-1} \vec{y}$.

[**] https://es.wikipedia.org/wiki/Distancia_de_Mahalanobis

$$= \left(\vec{x}_{(\cdot)} - \mathcal{Z}\vec{\beta}\right)^t V^{-1} \left(\vec{x}_{(\cdot)} - \mathcal{Z}\vec{\beta}\right)$$

con

$$\mathcal{Z} = \begin{bmatrix}\vec{1} & \vec{z}_{(\cdot)}\end{bmatrix} \qquad \vec{\beta} = \begin{bmatrix}\mu \\ \sigma\end{bmatrix}$$

La estimación de $\vec{\beta}$ es

$$\begin{aligned}
\hat{\beta} &= (\mathcal{Z}^t V^{-1} \mathcal{Z})^{-1} \mathcal{Z}^t V^{-1} \vec{x}_{(\cdot)} \\
\begin{bmatrix}\hat{\mu} \\ \hat{\sigma}\end{bmatrix} &= \begin{bmatrix}\vec{1}^t V^{-1}\vec{1} & \vec{1}^t V^{-1}\vec{m} \\ \vec{m}^t V^{-1}\vec{1} & \vec{m}^t V^{-1}\vec{m}\end{bmatrix}^{-1} \begin{bmatrix}\vec{1}^t \\ \vec{m}^t\end{bmatrix} V^{-1}\vec{x}_{(\cdot)} \\
&= \frac{\begin{bmatrix}\vec{m}^t V^{-1}\vec{m} & -\vec{1}^t V^{-1}\vec{m} \\ -\vec{m}^t V^{-1}\vec{1} & \vec{1}^t V^{-1}\vec{1}\end{bmatrix} \begin{bmatrix}\vec{1}^t \\ \vec{m}^t\end{bmatrix}}{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{m} - (\vec{1}^t V^{-1}\vec{m})^2} V^{-1}\vec{x}_{(\cdot)} \\
\langle \text{son escalares} \rangle &= \frac{\begin{bmatrix}\vec{m}^t V^{-1}\vec{m} & -\vec{m}^t V^{-1}\vec{1} \\ -\vec{1}^t V^{-1}\vec{m} & \vec{1}^t V^{-1}\vec{1}\end{bmatrix} \begin{bmatrix}\vec{1}^t \\ \vec{m}^t\end{bmatrix}}{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{m} - (\vec{1}^t V^{-1}\vec{m})^2} V^{-1}\vec{x}_{(\cdot)} \\
&= \frac{\begin{bmatrix}\vec{m}^t V^{-1}\vec{m}\vec{1}^t - \vec{m}^t V^{-1}\vec{1}\vec{m}^t \\ -\vec{1}^t V^{-1}\vec{m}\vec{1}^t + \vec{1}^t V^{-1}\vec{1}\vec{m}^t\end{bmatrix}}{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{m} - (\vec{1}^t V^{-1}\vec{m})^2} V^{-1}\vec{x}_{(\cdot)} \\
&= \frac{\begin{bmatrix}\vec{m}^t V^{-1}(\vec{m}\vec{1}^t - \vec{1}\vec{m}^t) V^{-1}\vec{x}_{(\cdot)} \\ \vec{1}^t V^{-1}(\vec{1}\vec{m}^t - \vec{m}\vec{1}^t) V^{-1}\vec{x}_{(\cdot)}\end{bmatrix}}{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{m} - (\vec{1}^t V^{-1}\vec{m})^2} \\
\hat{\mu} &= \frac{\vec{m}^t V^{-1}(\vec{m}\vec{1}^t - \vec{1}\vec{m}^t) V^{-1}\vec{x}_{(\cdot)}}{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{m} - (\vec{1}^t V^{-1}\vec{m})^2} \\
\hat{\sigma} &= \frac{\vec{1}^t V^{-1}(\vec{1}\vec{m}^t - \vec{m}\vec{1}^t) V^{-1}\vec{x}_{(\cdot)}}{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{m} - (\vec{1}^t V^{-1}\vec{m})^2}
\end{aligned}$$

Para distribuciones simétricas[††] se tiene $\vec{1}^t V^{-1}\vec{m} = 0$ y entonces

$$\hat{\sigma} = \frac{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{x}_{(\cdot)}}{\vec{1}^t V^{-1}\vec{1}\vec{m}^t V^{-1}\vec{m}} = \frac{\vec{m}^t V^{-1}\vec{x}_{(\cdot)}}{\vec{m}^t V^{-1}\vec{m}}$$

El estadígrafo W del contraste fue definido originalmente [5] como la estimación de la pendiente σ de la regresión arriba planteada multiplicada por una constante normalizadora (tipificadora):

$$W = \hat{\sigma}^2 \frac{(\vec{m}^t V^{-1}\vec{m})^2}{\vec{m}^t V^{-1} V^{-1}\vec{m}\, n\, S^2} = \frac{(\hat{\sigma}\vec{m}^t V^{-1}\vec{m})^2}{\vec{m}^t V^{-1} V^{-1}\vec{m}} \frac{1}{n\, S^2} = \frac{(\vec{m}^t V^{-1}\vec{x}_{(\cdot)})^2}{\|\vec{m}^t V^{-1}\|^2} \frac{1}{n\, S^2} = \left(\frac{\vec{m}^t V^{-1}}{\|\vec{m}^t V^{-1}\|}\vec{x}_{(\cdot)}\right)^2 \frac{1}{n\, S^2}$$

[††]En tal caso, $\vec{m}$ es simétrica, $m_i = -m_{n-i+1}$, y V es bisimétrica (simétrica y persimétrica), luego también lo es $V^{-1} = [v_{i,j}]_{i,j}$ y $v_{i,j} = v_{j,i} = v_{n-j+1,n-i+1} = v_{n-i+1,n-j+1}$. Entonces $\vec{1}^t V^{-1}\vec{m} = \sum_{i=1}^n \sum_{j=1}^n v_{i,j} m_j =$ $\sum_{j=1}^{\lfloor n/2 \rfloor} m_j \left(\sum_{i=1}^n v_{i,j} - \sum_{i=1}^n v_{i,n-j+1}\right) + [n \text{ impar}]\, m_{\frac{n+1}{2}} \left(\sum_{i=1}^n v_{i,\frac{n+1}{2}} - \sum_{i=1}^n v_{i,\frac{n+1}{2}}\right) =$ $\sum_{j=1}^{\lfloor n/2 \rfloor} m_j \left(\sum_{i=1}^n v_{i,j} - \sum_{i=1}^n v_{n-i+1,n-j+1}\right) + 0 = \sum_{j=1}^{\lfloor n/2 \rfloor} m_j \left(\sum_{i=1}^n v_{i,j} - \sum_{i=1}^n v_{i,j}\right) = 0$ (véase `https://math.stackexchange.com/questions/648856/how-to-prove-that-the-inverse-of-a-persymmetric-matrix-is-also-persymmetric`).

siendo $S^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2$ la varianza muestral. Habitualmente se expresa de alguna de las siguientes maneras:

$$\begin{aligned} W &= \frac{\left[\sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} a_{n-i+1}^{(n)}(X_{(n-i+1)} - X_{(i)})\right]^2}{nS^2} \\ &= \frac{\left[\sum_{i=1}^{n} a_i^{(n)}(X_{(i)} - \bar{X})\right]^2}{nS^2} \\ &= \frac{\left[\sum_{i=1}^{n} a_i^{(n)} X_{(i)}\right]^2}{nS^2} \end{aligned}$$

donde

$$\vec{a}^{(n)} = \begin{pmatrix} a_1^{(n)} \\ \vdots \\ a_n^{(n)} \end{pmatrix} = \frac{V^{-1}\vec{m}}{\sqrt{\vec{m}^t V^{-1} V^{-1} \vec{m}}}$$

Se verifican las siguientes propiedades:

- $a_i^n = -a_{n-i+1}^n$
- $0 < \frac{n a_1^2}{n-1} \leq W \leq 1$

Para su cálculo se puede usar una función de R o aproximar mediante Montecarlo:

```
> options(width=70)
> n <- 50; x <- 1:n
> shapiro.test (x)

        Shapiro-Wilk normality test

data:  x
W = 0.95558, p-value = 0.05809

> ## B <- 1e8 # repeticiones Montecarlo
> ### exige mucha memoria:
> ## Z <- t (replicate (1e8, sort(rnorm(n))))
> ## m <- colMeans (Z)
> ## V1 <- solve (var (Z))
> ### versión más ligera:
> ## m <- numeric (n)
> ## V <- matrix (0, n, n)
> ## for (i in 1:B)
> ## {
> ##     xi <- sort (rnorm (n))
> ##     m <- m + xi/B # medias
> ##     V <- V + tcrossprod (xi, xi) / B # medias de productos
> ## }
> ### para ahorrar tiempo
> ## save (m, V, file="shapiro.rda")
> load ("shapiro.rda") # ojo: sirve sólo para n=50
```

```
> V1 <- solve (V - tcrossprod (m, m)) # inversa de matriz de cov.
> a <- drop (V1 %*% m) / sqrt (drop (m %*% V1 %*% V1 %*% m))
> W <- function (x) sum (a * sort(x)) ^ 2 / sum ((x - mean(x)) ^ 2)
> W (x)

[1] 0.955301

> mean (replicate (1e5, W (rnorm (n))) <= W (x)) # P-valor

[1] 0.05719
```

6.4. Cramér - Von Mises y Anderson-Darling

Estas pruebas, ya vistas para la hipótesi nula simple $H_0\colon X \equiv F_0$, pueden adaptarse para la hipótesi compuesta de gausianidad $H_0\colon \exists\, \mu, \sigma,\ X \equiv \mathcal{N}(\mu, \sigma)$. Se usan los mismos estadígrafos que para H_0 simple pero sobre los datos tipificados mediante las estimaciones muestrales de media y desvío; luego se realiza un ajuste por tamaño muestral. Para CvM:

$$\begin{aligned} W &= \frac{1}{12n} + \sum_{i=1}^{n} \left[\Phi\left(\frac{x_{(i)} - \bar{x}}{s} \right) - \frac{2i-1}{2n} \right]^2 \\ W^* &= \left(1 + \frac{1}{2n} \right) W \end{aligned}$$

Para AD:

$$\begin{aligned} A &= -n - \frac{1}{n} \sum_{i=1}^{n} (2i-1) \left[\ln \Phi_{(i)} + \ln \Phi_{(n-i+1)} \right] \\ A^* &= \left(1 + \frac{3}{4n} + \frac{9}{4n^2} \right) A \end{aligned}$$

con $\Phi_{(i)} = \Phi\left(\frac{x_{(i)} - \bar{x}}{s} \right)$. Se usan W^* y A^* para calcular el P-valor, pero suelen mostrarse W y A:

```
> X <- 1:60
> nortest::cvm.test (X)

        Cramer-von Mises normality test

data:  X
W = 0.087, p-value = 0.1646

> nortest::ad.test (X)

        Anderson-Darling normality test

data:  X
A = 0.64345, p-value = 0.08892
```

7. Otras familias de distribuciones continuas

La idea de Lilliefors (emplear KS para hipótesi nula compuesta porque D_n no cambia por trasformaciones de ubicación y escala) puede emplearse con la familia gausiana (ya visto), la exponencial y la uniforme. Por ejemplo con $\mathcal{U}(0,\theta)$:

$$\begin{aligned} D_n &= \sup_{x\in\mathbb{R}}\left|\frac{\#\{X_i\leqslant x\}}{n}-F_{\hat\theta}(x)\right| = \sup_{0<x<\theta}\left|\frac{\#\{X_i\leqslant x\}}{n}-\frac{x}{X_{(n)}}\right| \\ &= \sup_{0<x<\theta}\left|\frac{\#\{X_i/\theta\leqslant x/\theta\}}{n}-\frac{x/\theta}{X_{(n)}/\theta}\right| = \sup_{0<y<1}\left|\frac{\#\{Y_i\leqslant y\}}{n}-\frac{y}{Y_{(n)}}\right| \end{aligned}$$

con $Y_i=\frac{X_i}{\theta}\overset{H_0}{\equiv}\mathcal{U}(0,1)$. Esa misma idea puede aplicarse también a los estadísticos de CvM y AD.

Para implementar estos contrastes, si no encontramos un paquete informático adecuado, podemos usar Montecarlo[‡‡]. Por comodidad, siguiendo la sugerencia de [6], se puede modificar el código de ciertas funciones de `R` para adaptarlo a un problema concreto. Por ejemplo, si quisiéramos usar CvM para hipótesi nula de distribución exponencial, modificaríamos en `nortest::cvm.test` el renglón

```
    p <- pnorm((x - mean(x))/sd(x))
```

por

```
    p <- pexp(x / mean(x))
```

Quedaría:

```
> cvm.exp <- function (x)
+ {
+     DNAME <- deparse(substitute(x))
+     x <- sort(x[complete.cases(x)])
+     n <- length(x)
+     if (n < 8)
+         stop("sample size must be greater than 7")
+     p <- pexp(x / mean(x))
+     W <- (1/(12 * n) + sum((p - (2 * seq(1:n) - 1)/(2 * n))^2))
+     WW <- (1 + 0.5/n) * W
+     if (WW < 0.0275) {
+         pval <- 1 - exp(-13.953 + 775.5 * WW - 12542.61 * WW^2)
+     }
+     else if (WW < 0.051) {
+         pval <- 1 - exp(-5.903 + 179.546 * WW - 1515.29 * WW^2)
+     }
+     else if (WW < 0.092) {
+         pval <- exp(0.886 - 31.62 * WW + 10.897 * WW^2)
+     }
+     else if (WW < 1.1) {
+         pval <- exp(1.111 - 34.242 * WW + 12.832 * WW^2)
+     }
+     else {
```

[‡‡] Como en la página 29.

```
+         warning("p-value is smaller than 7.37e-10,
+                 cannot be computed more accurately")
+         pval <- 7.37e-10
+     }
+     RVAL <- list(statistic = c(W = W), p.value = pval,
+                 method = "Cramer - Von Mises exponencial",
+         data.name = DNAME)
+     class(RVAL) <- "htest"
+     return(RVAL)
+ }
```

La aproximación del P-valor mediante Montecarlo podría hacerse así:

```
> X <- 1:20
> n <- length(X)
> B <- 1e4 # número de iteraciones
> distriW <- replicate (B,
+                       cvm.exp (rexp (n)) $ statistic)
> mean (distriW >= cvm.exp (X) $ statistic)

[1] 0.026
```

La distribución cumple con el tamaño del contraste para cualquier exponencial:

```
> ## RC = [W > k]
> alfa <- 0.05
> k <- quantile (distriW, 1-alfa)
> ## vale para cuaquier valor de landa
> (landa1 <- runif (1, 0, .1))

[1] 0.05660434

> (landa2 <- runif (1, 100, 200))

[1] 145.2118

> mean (replicate (B, cvm.exp (rexp (n,landa1)) $ statistic) > k)

[1] 0.0505

> mean (replicate (B, cvm.exp (rexp (n,landa2)) $ statistic) > k)

[1] 0.051
```

Ojo, porque el P-valor devuelto directamente por `cvm.exp` puede alejarse bastante, ya que su cálculo está implementado específicamente para la distribución gausiana:

```
> nortest::cvm.test (X)
```

```
        Cramer-von Mises normality test

data:  X
W = 0.029002, p-value = 0.8511


> cvm.exp (X)

        Cramer - Von Mises exponencial

data:  X
W = 0.2619, p-value = 0.0007799
```

Referencias

[1] https://doi.org/10.1214/aoms/1177730256

[2] https://luk.staff.ugm.ac.id/stat/ks/Kolmogorov-SmirnovDTable.pdf

[3] J.D. Gibbons y S. Chakraborti. Nonparametric statistical inference. 1992. ISBN 0-8247-8661-0.

[4] Blom, G. (1958) Statistical Estimates and Transformed Beta Variables. Wiley.

[5] Shapiro y Wilk (1965) https://sci2s.ugr.es/keel/pdf/algorithm/articulo/shapiro1965.pdf

[6] https://stats.stackexchange.com/questions/111978/anderson-darling-exponential-distribution